

oVirt 3.4

Arho Vartiainen

Opinnäytetyö

Syyskuu 2014

Tietoverkkotekniikan koulutusohjelma

Tekniikan ja liikenteen ala



JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES



Tekijä(t) Arho Vartiainen	Julkaisun laji Opinnäytetyö	Päivämäärä 24.9.2014
	Sivumäärä 48	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty (X)
Työn nimi oVirt 3.4		
Koulutusohjelma Tietotekniikka		
Työn ohjaaja(t) Rantonen, Mika; Häkkinen, Antti		
Toimeksiantaja(t) Jyväskylän ammattikorkeakoulu, Hallberg, Jani		
<p>Tiivistelmä</p> <p>Opinnäytetyön tarkoituksena oli tutkia avoimen lähdekoodin Ovirt-virtualisointiohjelmistoa. Ohjelmiston mahdollisia hyötyjä ja etuja tutkittiin sekä pohdittiin olisiko siitä hyötyä opiskelijoille mahdollisesti tarjottavassa palvelinvirtualisointi opintojaksossa. Lisäksi ohjelmiston tarjoamia ominaisuuksia tutkittiin ja arvioitiin käytännön osuudessa. Tutkimusten tulosten osalta tehtiin päätelmiä toimivatko ominaisuudet käytännössä ja onko ohjelmistosta hyötyä opiskelijoille.</p> <p>Lisäksi opinnäytetyön aiheena olevaa Ovirt virtualisointikokonaisuutta vertailtiin hieman sen suoraan kilpailijaan, VMware vSphereen, ja tehtiin päätelmiä onko Ovirt suoraan kelpoinen korvaajaksi tai mahdolliseksi rinnakkaiseksi virtualisointialustaksi.</p> <p>Työ toteutettiin kahdella työasemalla joiden päälle virtualisoitiin tarvittava määrä palvelimia Ovirtin testaamista varten. Ovirt-virtualisointialusta pystytettiin näille virtuaalikoneille sekä Ovirtin toiminnallisuutta testattiin esitetyihin ominaisuuksiin nähden.</p>		
Avainsanat (asiasanat) oVirt, virtualisointi, palvelinympäristö		
Muut tiedot		



Author(s) Arho Vartiainen	Type of publication	Date
	Bachelor's thesis	24.9.2014
	Pages 48	Language Finnish
		Permission for web publication (X)
Title oVirt 3.4		
Degree programme Information technology		
Tutor(s) Rantonen, Mika; Häkkinen, Antti		
Assigned by JAMK University of Applied Sciences, Hallberg, Jani		
<p>Abstract</p> <p>The purpose of this bachelor's thesis was to study the Ovirt virtualization software and try to discern if it can be used as a tool for students in the possibly upcoming server virtualization course. The advertised features of the software were studied and reviewed, deductions were made from these studies if the features are functional in real world and whether Ovirt is useful for students.</p> <p>In addition, Ovirt was compared with the direct competitor, VMware vSphere, and there were some deductions on Ovirt being able to replace VMware vSphere or another solution alongside the vSphere for virtualization platform.</p> <p>The practical part of the thesis was carried out using two workstations. Ovirt virtualization platform and the necessary servers were virtualized on top of these workstations and Ovirt's functionalities were tested against the advertised features of Ovirt.</p>		
Keywords oVirt, virtualization		
Miscellaneous		

Sisältö

Käsitteitä	4
1 Työn lähtökohdat	7
1.1 Tausta	7
1.2 Tavoitteet	8
1.3 Tilaaaja	8
2 Virtualisointi	9
2.1 Historia	9
2.2 Peruskäsitteet	9
2.2.1 Yleistä	9
2.2.2 Virtualisoinnin toteutustavat	10
2.2.3 Paravirtualisointi	11
2.2.4 Täysi virtualisointi	12
2.2.5 Osittainen virtualisointi	12
2.3 Virtualisoinnin edut	12
3 Ovirt-arkkitehtuuri	13
3.1 Yleistä	13
3.2 Management	13
3.3 Libvirt	14
3.4 KVM	15
3.5 QEMU	16
3.6 VDSM	16
3.7 SPM	16

4	Ovirt	17
4.1	Red Hat	17
4.2	Ovirt	18
4.3	Ovirtin ominaisuuksia	18
5	Levyjärjestelmät	21
5.1	iSCSI	22
5.2	GlusterFS	22
5.3	NFS	23
5.4	Muut levytila ratkaisut	23
6	FreeIPA	24
7	Käytäntö	24
7.1	Manageri	26
7.2	Levytila	27
7.3	Isäntäkoneet	29
7.4	FreeIPA	30
7.5	Kokonaisuus	30
8	Vertailua vSphereen	34
9	Päätelmät	36
	Lähteet	41
	Liitteet	42
	Management	42
	iSCSI-levysäiliö	43
	Ovirt isännät	43
	FreeIPA	43

GlusterFS	44
Ovirtin käyttö	46

Kuviot

Kuvio 1	Virtualisointityypit	10
Kuvio 2	Suojarengasmalli	11
Kuvio 3	Ovirtin arkkitehtuuri	14
Kuvio 4	Libvirt	15
Kuvio 5	VDSM arkkitehtuuri	17
Kuvio 6	Yleiskuva Ovirtistä	19
Kuvio 7	Verkkorajapinnan manipulointi Ovirtin hallinnassa	21
Kuvio 8	FreeIPAn päänäkömä Admin käyttäjälle	24
Kuvio 9	Virtuaalikoneiden jakautuminen fyysisille koneille.	25
Kuvio 10	Ovirtin hallintapaneeli hostien ja tallenustilan lisäyksen jälkeen	27
Kuvio 11	Centos palvelimen iSCSI jako	28
Kuvio 12	Centos iSCSI palvelimen IPtables säännöt	28
Kuvio 13	Live migration siirto hostilta toiselle	31
Kuvio 14	Pingin hyvin pieni nousu siirron aikana	32
Kuvio 15	Templaten luonti Ovirtissä	33
Kuvio 16	Normaalikäyttäjän näkymä kirjautuessa	33
Kuvio 17	Virtuaalikoneen käynnistys peruskäyttäjällä	34
Kuvio 18	Snapshotteja virtuaalikoneista vasemmassa alakulmassa	37

Käsitteitä

AAA	Authentication, Authorization & Accounting. Protokolla, jonka avulla käyttäjä voidaan tunnistaa ja antaa resursseja käytettäväksi.
AMD-V	Advanced Micro Dynamicsin virtualisointilaajennokset suorittimelle.
API	Application Programming Interface. Ohjelmointirajapinta, jonka avulla erilliset ohjelmat voivat ”keskustella” keskenään.
BSD	Berkeley Software Distribution. Unix käyttöjärjestelmä.
DNS	Domain Name System. Nimipalvelujärjestelmä, joka muuntaa verkotunnuksia IP-osoitteiksi.
Guest	Virtualisoidusta koneesta käytetty termi.
Host	Virtualisointialusta guest koneille.
Hypervisor	Laitteisto tai ohjelmisto, joka hallitsee virtuaalikoneita.
Intel VT-x	Intelin virtualisointilaajennokset suorittimen käskykannalle.
IP	Internet Protocol. Yksi Internetissä käytettävä osoitteistus joilla tietokoneet voivat kommunikoida toistensa kanssa.
iSCSI	Internet small computer system interface. IP-pohjainen verkkostandardi tallennustilan jakamiselle tietoverkon yli.
Kernel	Käyttöjärjestelmän ydin. Hoitaa kommunikoinnin fyysiselle laitteistolle.

Kytkin	Tietoverkkolaite, joka operoi yleensä OSI-mallin toisella kerroksella toimittaen datapaketteja laitteille MAC-osoitteiden perusteella.
LUN	Logical Unit Number. iSCSI:n tapauksessa määritelty levy, joka jaetaan sitä haluaville asiakaskoneille.
MAC	Media Access Control Address. Verkkolaitteiden rajapintojen identifiointiin suunniteltu osoitteistus. Osoite koostuu laitevalmistajan ja rajapinnan uniikista osasta, ja sitä käytetään OSI-mallin toisen kerroksen viestintään.
NAT	Network Address Translation. Osoitteenmuunnos verkkotekniikassa, jolla voi mm. piilottaa useita koneita yhden (julkisen) IP-osoitteen taakse.
OS	Operating System. Yleinen lyhenne käyttöjärjestelmästä.
OSI	Open System Interconnection. Seitsemästä kerroksesta koostuva malli, jolla pyritään standardisoimaan verkkoliikenteen toimintaa. Kerrokset ovat 1. Fyysinen kerros, 2. Siirtokerros (MAC), 3. Verkkokerros (IP), 4. Kuljetuskerros (TCP,UDP), 5. Istuntokerros, 6. Esitystapakerros ja 7. Sovelluskerros.
POSIX	Portable Operating System Interface. Kokoelma standardeja jotka määrittävät rajapinnan ohjelmien ja käyttöjärjestelmän välillä.
RDP	Remote Desktop protocol. Microsoftin etätyöpöytäprotokolla. Sallii työpöydän jakamisen verkon yli.
SSH	Secure Shell. Salattuun tietoliikenteeseen tarkoitettu protokolla, jolla voi ottaa (merkkipohjaisen) etäyhteyden toiseen koneeseen.

TCP	Transmission Control Protocol. OSI-mallin neljännen kerroksen protokolla, jonka yhteydet ovat luotettavia. Käytössä suuressa osassa internetliikenteessä.
UDP	User Datagram Protocol. Myös OSI-mallin neljännen kerroksen protokolla, jonka yhteydet eivät ole luotettavia. Käytössä usein reaaliaikasovelluksissa kuten videon siirrossa.
VM	Virtual Machine. Yleinen lyhenne virtualisoidusta tietokoneesta.
x86	Proessorien arkkitehtuuri ja käskykanta. Käytössä nykyisissä pöytätietokoneissa ja palvelimissa.

1 Työn lähtökohdat

1.1 Tausta

Opinnäytetyön aiheena oli avoimen lähdekoodin Ovirt-virtualisointikokonaisuus. Ovirt-virtualisointikokonaisuudesta tehtiin johtopäätöksiä, voisiko sitä käyttää mahdollisesti opetukseen tai muihin tarkoituksiin Jyväskylän ammattikorkeakoulussa. Ohjelmisto on rakennettu pienemmistä avoimen lähdekoodin osista, joista muutamaiset ovat olleet käytössä pidemmänkin ajan. Ovirtin avulla osat on yhdistelty yhdeksi kokonaisuudeksi.

Ovirtin versioksi valikoitui kirjoitushetkellä 3.4, koska se oli uusin. Vanhempaa versiota olisi voinut myös tutkia, mutta tällöin järkevämpää olisi ollut ottaa käyttöön Red Hat Enterprise Virtualization tuote testikokeiluun, koska se on huomattavasti kattavammin dokumentoitu ja mahdollisissa ongelmatapauksissa on tukea saatavilla Red Hatilta maksua vastaan. Red Hat Enterprise Virtualization on mahdollista ladata ja asentaa Red Hatin sivuilta, ja ohjelmistoa on mahdollista käyttää 60 päivää maksutta.

Koska Ovirt on vielä suhteellisen tuore ratkaisu, haasteena oli lähdemateriaalin laatu ja saatavuus. Ovirtin versiolle 3.4 ei ollut painettua kirjallisuutta kirjoitushetkellä vielä lainkaan. Tästä syystä johtuen kaiken lähdemateriaalin joutui tutkimaan sähköisinä julkaisuina blogi- ja wikityyppisten kirjoitusten muodossa. Ovirtin wikisivuilla oli myös hieman dokumentaatiota ja yhtenä lähteenä oli myös Red Hatin dokumentaatiot, joita sovellettiin Ovirtiin. Vanhemmille 3.3 versioille oli yksi kirja saatavana, mutta lainattavaksi ei kirjastoista löytynyt yhtään.

Käytännön osuudessa tehdyissä tutkimuksissa oli käytössä tilaajan puolelta kaksi työasemaa, joiden päälle oli tarkoitus virtualisoida Ovirt-kokonaisuus. Tilaajan puolelta käyttöjärjestelminä oli asennettu Windows 7:t ja näihin oli asennettu VMware Workstation 9-ohjelmat. Näin toimimalla pienestä laitemäärästä oli tarkoitus saada riittävästi virtuaalikoneita Ovirtin testaamiseen. Koneiden määrittelyistä tarkemmin luvussa 7.

1.2 Tavoitteet

Tavoitteena oli saada asennettua ja testattua ohjelmistoa niin, että ohjelmistolla voisi mahdollisesti testata ja toteuttaa oppimisympäristössä virtualisointia. Nyt joillakin opintojaksoilla hieman tutuksi tullut VMwaren vSphere olisi mahdollista siirtää sivuun ja säästää lisenssimaksuissa tai laittaa opiskelijat rakentamaan itselleen oma pilvipalvelualusta. Yksi vaihtoehto olisi myös pitää Ovirtia rinnakkain vSpheren kanssa, mutta tämä lisäisi tarvittavan laitemäärän käytännössä kaksinkertaiseksi.

Tämän lisäksi Ovirt on myös avoimen lähdekoodin kokonaisuus, joten ohjelmointiin suuntautuvat opiskelijat voisivat helposti saada haltuunsa lähdekoodin ja toteuttaa siihen uusia toimintoja ja yrittää saada niitä integroitavaksi tuleviin julkaisuihin.

1.3 Tilaaja

Tilaajana opinnäytetyölle toimi Jyväskylän ammattikorkeakoulun teknologian yksikkö. Teknologiayksikkö jakautuu viiteen tulosalueeseen: IT-instituutti, konetekniikka, logistiikka, rakentaminen ja biotalousinstituutti. Opinnäytetyö tehtiin IT-instituutin puolelle Dynamon kampukselle ja tilaajaksi merkittiin laboratorioinsinööri Jani Hallberg. IT-instituutissa on koulutustarjonnasta mahdollista valita itselleen mediatekniikan, ohjelmistotekniikan tai tietotekniikan koulutusohjelma ja koulutusta on mahdollista jatkaa vielä ylempään ammattikorkeakouluun jossa tarjolla on informaatioteknologian (information technology) koulutusohjelma.

Jyväskylän ammattikorkeakoulussa on opiskelijoita yli 8500 ja koulu työllistää yli 700 henkilöä. Kampuksia Jyväskylän ammattikorkeakoululla mainitun Lutakossa olevan Dynamon toimipisteen lisäksi on Rajakadun ja Puistokadun pääkampukset, Pitkäkadun Suomalainen musiikkikampus ja Biotalousinstituutti Saarijärvellä (Jyväskylän ammattikorkeakoulu 2014).

2 Virtualisointi

2.1 Historia

Virtualisointi nimenä sai alkunsa 1960- ja 1970-luvuilla, kun tarpeena oli jakaa kalliiden mainframe-tietokoneiden laitteistoresurssit hyötykäyttöön eri ihmisten välille.

Vuonna 1967 kehitettiin ensimmäinen hypervisor CP-40 käyttöjärjestelmälle, joka oli suunnattu System/360 koneita varten. Heti seuraavana vuonna toinen versio nimeltään CP-67 julkaisiin. Jälkimmäisen hypervisor salli muistin jakamisen virtuaalikoneiden välillä ja jokainen käyttäjä sai käyttöön oman muistiavaruuden (Milberg 2009). Erona nykyisiin virtualisointiratkaisuihin ja hypervisoreihin nämä vanhat toteutukset olivat rautatason virtualisointeja (firmware).

Vuonna 1999 virtualisointi saavutti niin sanotusti suuren yleisön, kun VMwarella onnistuttiin ensimmäisenä virtualisoimaan x86 arkkitehtuurin koneita (VMware 2007). Kyseinen tapahtuma onkin tietotekniikassa johtanut erittäin merkittäviin muutoksiin erityisesti palvelinpuolella, jossa virtualisointi on jokapäiväistä toimintaa.

2.2 Peruskäsitteet

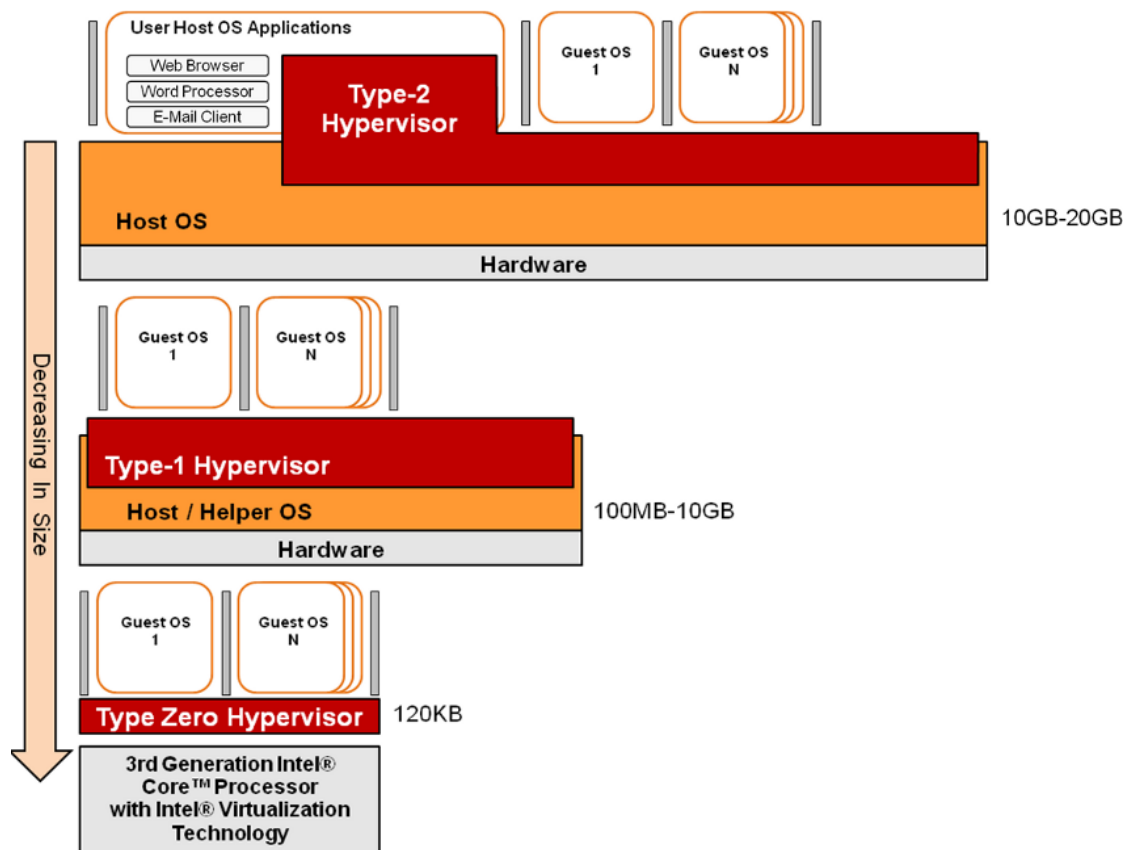
2.2.1 Yleistä

Virtualisointi tietotekniikassa on kattotermi, joka sisältää sisäänsä monia erilaisia toteutuksia ja mahdollisuuksia. Virtuaalinen tietokone on fyysisen koneen tai laitteen ohjelmallisesti tehty looginen ilmentymä. Virtualisoinnilla saadaan yhden fyysisen koneen päälle rakennettua useita erillisiä virtuaalisia tietokoneita. Tämä johtaa tilanteeseen, jossa yhden koneen resurssit saadaan hyötykäyttöön tehokkaasti (IBM 2007).

Hypervisorit on perinteisesti ottaen jaettu kahteen eri tyyppiin. Tyypin yksi virtualisointi tapahtuu suoraan raudalle (baremetal) asennettavan suhteellisen vähän laitteistoresursseja kuluttavan hypervisorin avulla. Tyypin kaksi virtualisointi taas tehdään jo pohjilla olemassa olevan käyttöjärjestelmän (GNU/Linux, Windows, OSX...) päälle, joka huolehtii virtualisoinnista jonkin

ohjelmiston avulla (VirtualBox, VMware workstation, Parallels...). Joidenkin mielestä jako ei ole enää riittävä, joten he haluaisivat mukaan vielä tyypin nolla hypervisorin, mutta yleistä käytäntöä asiasta ei vielä ole (Keegan 2013). Kuviossa 1 on esillä, kuinka hypervisorit eroavat toisistaan.

Riippumatta tavasta, jolla tietokone tai laitteisto virtualisoidaan, tämä aiheuttaa rajoitteita sekä laitteistolle että ohjelmistoille. Ongelmattoman toiminnan jatkumiseksi virtuaalinen tietokone ei voi käyttää laitteistoresursseja suoraan, vaan sen on tehtävä pyynnöt laitteistoresursseille alla olevan hypervisorin kautta.



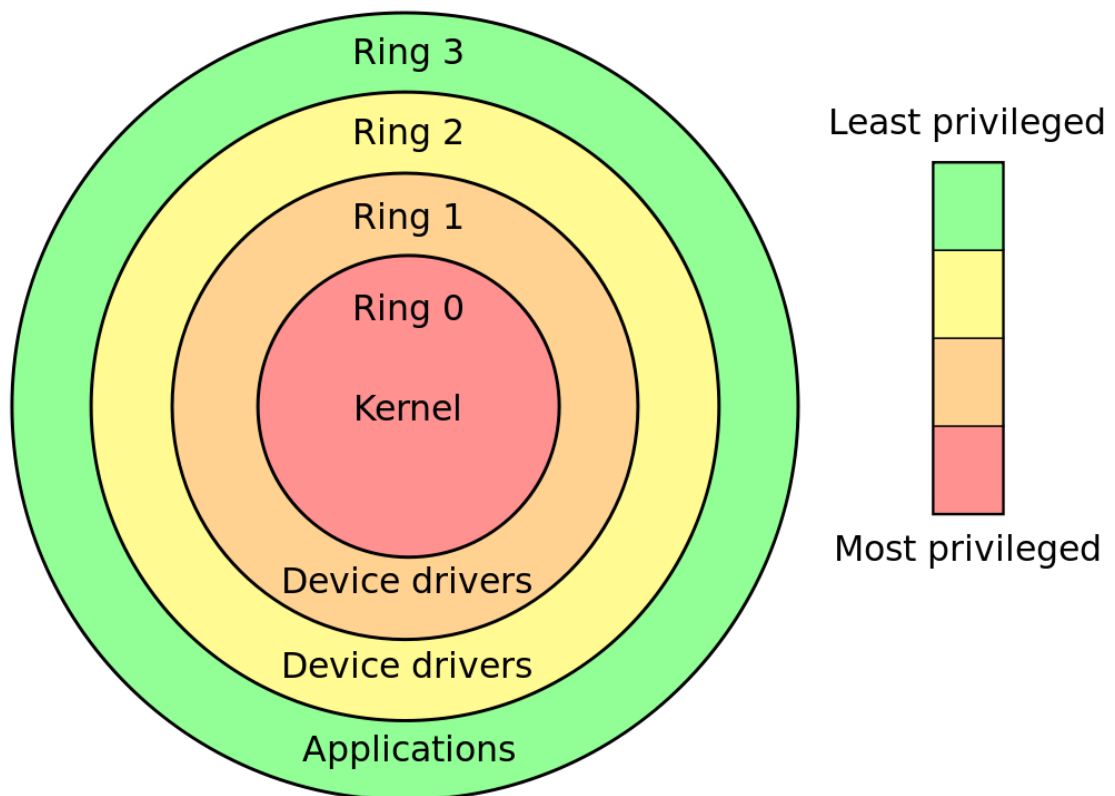
Kuvio 1: Virtualisointityypit (alkup. kuvio Keegan 2013)

2.2.2 Virtualisoinnin toteutustavat

Koska käyttöjärjestelmät on usein luotu toimimaan suoraan raudalla, tekevät ne suoria pyyntöjä alla olevalle raudalle. Tämä aiheuttaa ongelmatilanteen laitteistoresursseille. Jos virtuaalikoneet tekevät yhtäaikaista pyynnön jollekin laiteresurssille, se voi jäädä tilaan, joka ei ole hallittu. Näin tapahtuessa voi koko kone kaatua (Ribeiro 2009). Tätä varten onkin nykyiset käyttöjärjestelmät ja laitearkkitehtuuri (x86) suunniteltu niin, että käyttöjärjestelmät käyttävät

suojarahengasmallia (ks. kuvio 2) pohjanaan. Käyttöjärjestelmän ydintä ajetaan suojarahenkaan kehällä 0 ja käytettäviä ohjelmia yleensä kehällä 3. Kehämallissa ohjelmat eivät voi suoraan käyttää laiteresursseja, vaan niiden täytyy tehdä pyyntö alemmalle kehälle. Tämän ansiosta ei ohjelman vääränlainen toiminta yleensä kaada koko käyttöjärjestelmää, vaan ainoastaan ohjelman toiminta voi lakata.

Täyden virtualisoinnin ratkaisuihin, kun mukana on laitteistoavustus (Intel VT tai AMD-V), luodaan hypervisorin varten rengas -1. Nyt virtualisoidut käyttöjärjestelmät toimivat edelleen normaalisti renkaalla 0, mutta alapuolelta löytyy hypervisor renkaalta -1.



Kuvio 2: Suojarahengasmalli (alkup. kuvio Wikipedia)

Virtualisointia voi tehdä muutamalla eri tavalla. On niin kutsuttua paravirtualisointia, täyttä virtualisointia ja osittaista virtualisointia, joista jokaisella on etunsa ja haittansa toisiinsa nähden.

2.2.3 Paravirtualisointi

Paravirtualisoinnissa vieraskäyttöjärjestelmän ydintä muokataan niin, että se ei toimi suoraan suojarahengasmallin kehällä 0, vaan siirtää itsensä seuraaville kehille.

Usein kaupallisten käyttöjärjestelmien kanssa paravirtualisointia ei voi tehdä, koska ydintä ei ole annettu vapaasti jakoon, mutta erinäisten Linux- ja BSD-varianttien kanssa tämä onnistuu, koska ytimen lähdekoodit ovat saatavilla ja siten muokattavissa.

2.2.4 Täysi virtualisointi

Täydessä virtualisoinnissa koneet virtualisoidaan täysin niin, että virtuaalisella tietokoneella on hypervisorin toteuttamat virtuaalikomponentit (prosessori, levyt, verkkorajapinnat jne.). Täyttä virtualisointia on kahta mallia: laitteistoavusteista ja avustamatonta. Laitteistoavusteissa virtualisoinnissa tietokoneen suorittimella on muutama ylimääräinen käsky, jolloin virtuaalikoneen suoritus tapahtuu lähes samalla vaivalla kuin ilman virtualisointia. Avustamattomassa mallissa taas virtuaalikoneet pyytävät suorittimelta toiminteita, joita täytyy tulkata. Tämä saattaa johtaa tilanteeseen, jossa merkittävä osa suorituskyvystä hukataan tulkkaamiseen.

2.2.5 Osittainen virtualisointi

Osittaisessa virtualisoinnissa taas vieraskäyttöjärjestelmät käyttävät isäntäkoneen ydintä (kernel) ja vain osa toiminteista on virtualisoituja. Tunnetuimpia esimerkkejä tästä ovat FreeBSD:n Jails ja Linux-säiliöt (LXC, Linux Containers).

Osittaisen virtualisoinnin etuihin kuuluu suoraan raudalle asennetun käyttöjärjestelmän nopeus, mutta suurimpana ongelmana virtualisoitujen säiliöiden täytyy olla samaa käyttöjärjestelmää. Täten Linuxien päällä ei voi ajaa Windows-käyttöjärjestelmiä ja toisinpäin.

2.3 Virtualisoinnin edut

Virtualisoinnin etuna voidaan pitää pienentyntä sähkönkulutusta. Lisäksi palvelinsaleista saadaan kertaluokkaa pienempiä, koska fyysisiä koneita tarvitaan vähemmän samaan määrään käyttöjärjestelmien asennusten kanssa. Lisäksi virtualisoidut koneet on helppo siirtää toiselle alustalle esimerkiksi laiterikon sattuessa tai jos on tarve vaihtaa virtualisointialustaa.

Vaikka virtuaalikoneet pyörivätkin saman raudan päällä, eivät virtuaaliset koneet tiedä toisistaan mitään ilman erillisiä ohjelmistoja ja määrittäjiä. Tästä suorana etuna on tietoturvan lisääntyminen, koska yhden virtualikoneen tietomurron seurauksena eivät muut koneet ole automaattisesti vaarassa. Lisäksi yhden virtuaalisen koneen rikkoutuminen ei vaikuta hypervisorin päällä pyöriviin muihin koneisiin, koska vika rajoittuu ainoastaan vikaantuneen virtuaalikoneen määrittäjiin tai ohjelmistoihin.

3 Ovirt-arkkitehtuuri

3.1 Yleistä

Ovirt koostuu useammasta avoimen lähdekoodin komponentista.

Peruspalikat Ovirtille ovat hallinta, hostit ja levytila. Näihin kaikkiin on avoimen lähdekoodin komponentit, joilla toiminta saadaan nivottua yhdeksi kokonaisuudeksi. Toiminnan kannalta ensimmäisenä pystytetään management, joko suoraan raudalle tai virtualisoituna Ovirt-pilveen (niinkutsuttu self-hosted engine). Seuraavaksi hostit lisätään pilvelle käytettäväksi ja tämän jälkeen lisätään tallennustila. Kun nämä kaikki ovat lisätty, on hallintakoneelta mahdollista lisätä ja poistaa virtuaalikoneita halutuun määritykseen.

Luvuissa x-y puhutaan Ovirtin taustalla olevista komponenteista ja siitä, mitä ne tekevät ja miten ne liittyvät Ovirtin toimintaan.

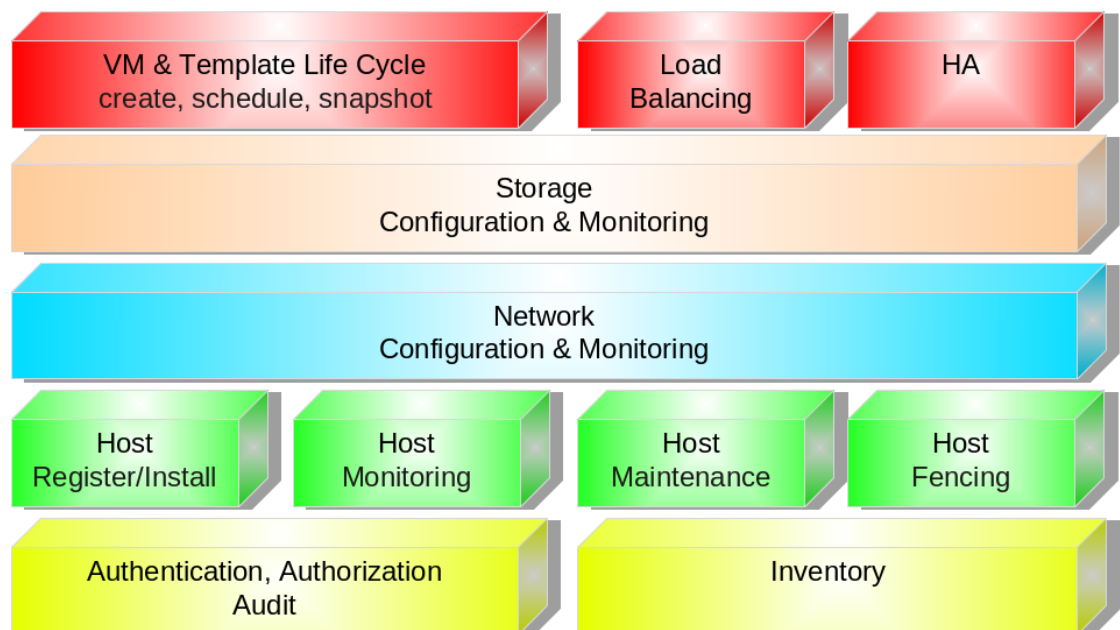
3.2 Management

Ovirt management on web-käyttöliittymä pilven hallintaan. Hallinnan avulla on mahdollista muun muassa lisätä isäntäkoneita, luoda tai poistaa virtuaalikoneita ja siirtää niitä eri hostien välillä. Koska toiminteita on runsaasti, on hallintaan parempi tutustua itse asentamalla vaikkapa niin kutsutun AIO-asennuksen (All in one), joka sisältää kaiken tarvittavan testaamista varten.

Hallinnasta voi myös antaa käyttäjille oikeudet hallita joko tiettyjä isäntäkoneita, johon voi luoda virtuaalikoneita, tai antaa käyttäjälle oikeudet ainoastaan virtuaalikoneiden hallintaan. Kummassakin tapauksessa käyttäjää ei

managementin avulla voi luoda, vaan siihen tarvitaan erillistä palvelinta. Tämä voi olla esimerkiksi Windows palvelimen Active Directory tai avoimen lähdekoodin puolelta FreeIPA-palvelin. Hallinnan pääkäyttäjän päänäköymään voi tutustua kuvista 10.

Management sitoo kaikki myöhemmin mainitut komponentit yhdeksi kokonaisuudeksi. Ovirt managementtiä ja sen toimintoja havainnollistava kuvio 3 antaa selkeän perusidean ohjelmiston kattavuudesta ja siitä, mitä kaikkea koko Ovirt-kokonaisuudella voi toteuttaa. Kuten kuvista 3 voidaan nähdä, management koneen vastuulla ovat infrastruktuurin osalta hallinnalliset toimet pilvestä. Alimmasta ylimpään järjestyksessä management-koneen vastuulla on tunnistautuminen erillistä palvelinta vasten (authentication, authorization, audit), mitä kaikkea Ovirt-pilveen on lisätty (inventory), Ovirt-isäntäkoneiden lisääminen, monitorointi, huoltaminen. Lisäksi management-koneella luodaan verkko- ja tallennustilakonfiguraatiot sekä määrittellään virtuaalikoneiden pohjavedokset, kuorman jakaminen ja korkea saatavuus.

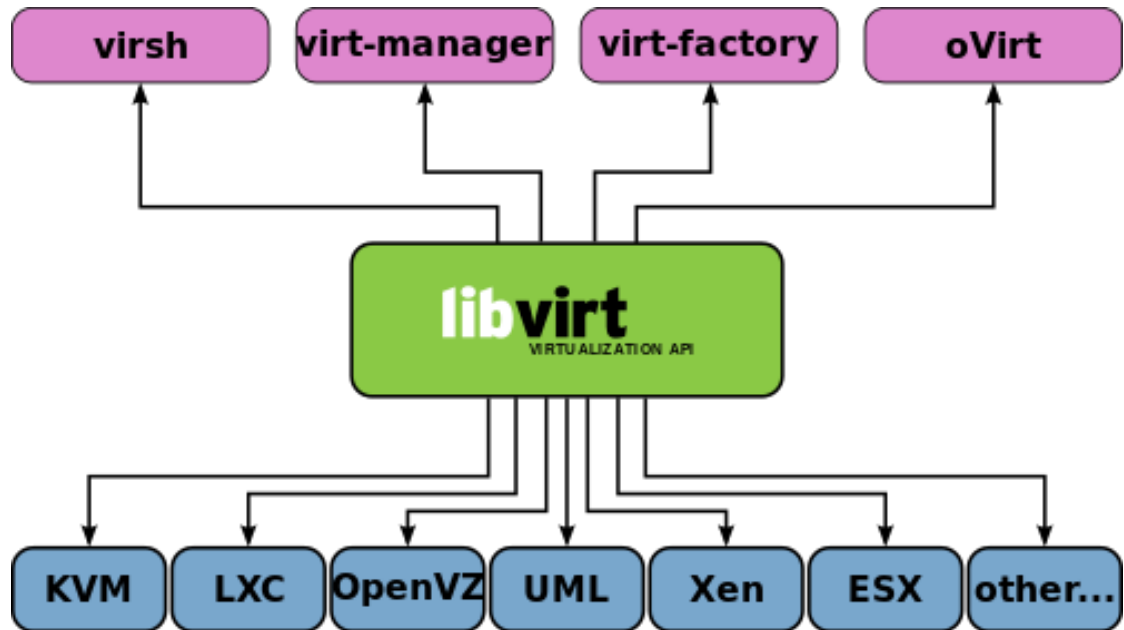


Kuvio 3: Ovirtin arkkitehtuuri (alkup. kuvio Ovirt 2010)

3.3 Libvirt

Libvirt on kirjasto ja API (application programming interface), joka toteuttaa pyynnöt virtuaalikoneiden käynnistämisestä ja sammuttamisesta sekä huolehtii virtualisoiduista osista (suorittimet jne.) ja välittää tästä tiedon ylemmälle

kerrokselle (ks. kuvio 4). Ylempänä kerroksena Ovirtin tapauksessa on aluksi hostilla pyörivä VDSM, joka keskustelelee Ovirt managerin kanssa.



Kuvio 4: Libvirt (alkup. kuvio Wikipedia)

3.4 KVM

KVM (Kernel-based Virtual Machine) on virtualisointikerros Linuxin ytimeen (kernel), jolla Linux-ytimellä varustetusta tietokoneesta saa luotua niin kutsutun hypervisorin. KVM vaatii suorittimelta joko Intel VT tai AMD-V -laajennukset toimiakseen.

Hypervisor hallitsee asiakaskäyttöjärjestelmiä (guest OS) antaen näille resurssit käyttöön virtualisoituina moduuleina (prosessorit, verkkokortit, kiintolevyt jne.). KVM:n tapauksessa KVM antaa QEMUlle (Quick Emulator, tästä lisää seuraavassa osiossa) mahdollisuuden käyttää järjestelmän fyysistä rautaa niin, että yksittäistä QEMU:n instanssia suoritetaan käyttäjätasolla (ei ytimessä suoraan) (Burden et al. 2014).

KVM on hieman erikoinen hypervisor, koska virtualisointi tapahtuu tyypin yksi mukaan (bare metal), mutta kuitenkin niin, että sen saa käyttöön myös käyttöjärjestelmän sisältä (tyyppi 2). KVM siis ikäänkuin ottaa vallan pohjalla olevalta käyttöjärjestelmältä (Day 2012).

3.5 QEMU

QEMU (Quick Emulator) on alustariippumaton (multiplatform) emulaattori, jolla voi tehdä täyden järjestelmän emuloinnin. QEMU emuloi täyttä järjestelmää, kuten PC:tä (Personal Computer), joka voi sisältää yhden tai useamman suorittimen ja oheislaitteet. QEMUlla voidaan käynnistää useita erilaisia käyttöjärjestelmiä tai debugata järjestelmäkoodia. QEMU yhdessä KVM:n ja suorittimen lisäoptioiden kanssa tarjoaa täyden laitteistokiihdytetyn virtualisoinnin (Burden et al. 2014).

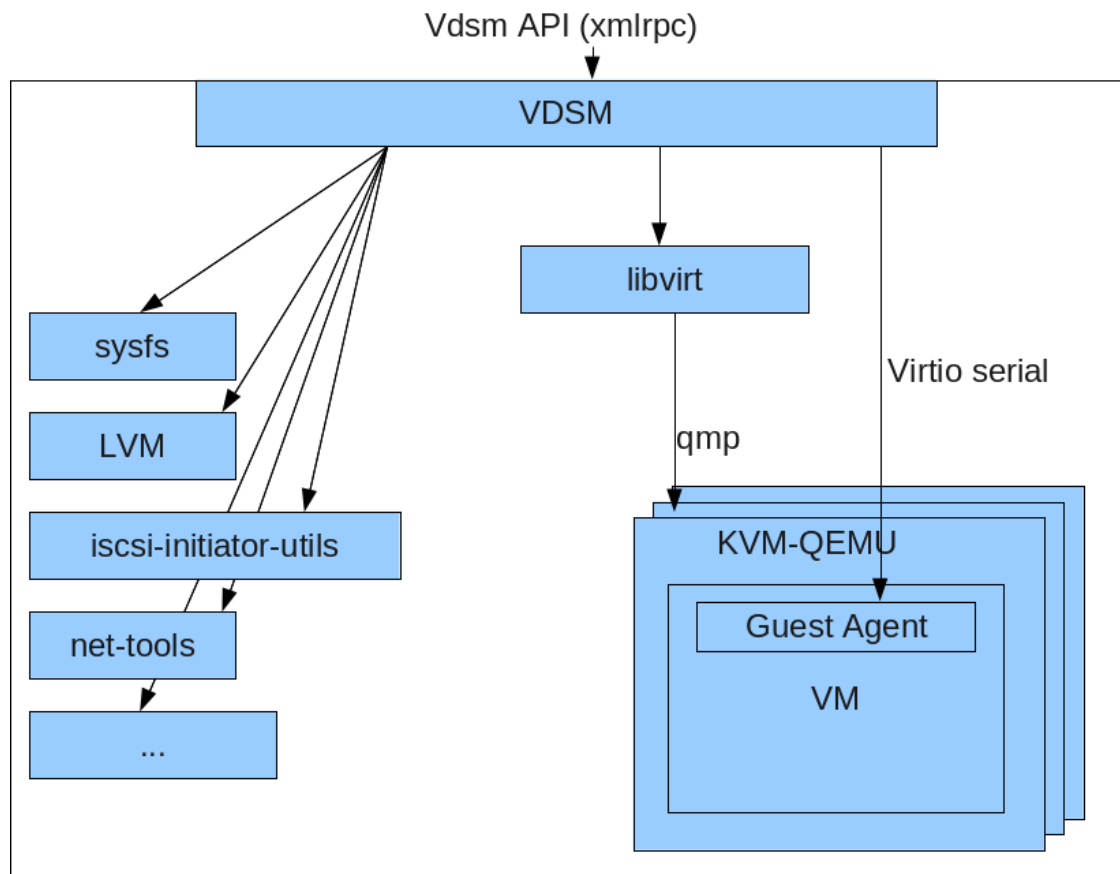
Tietokoneen voi virtualisoida täysin myös ilman KVM kernel-moduulia, mutta virtuaalisen tietokoneen toiminta on mitä todennäköisimmin äärimmäisen hidasta.

3.6 VDSM

VDSM (Virtual Desktop and Server Manager) on rajapinta Ovirt managerin ja isäntäkoneiden kanssa. Ovirt antaa käskyjä hosteille VDSM-rajapinnan yli lisäten niille muun muassa verkkoja käytettäväksi virtuaalikoneita varten. VDSM myös kerää tietoa hostin suorituskyvystä sekä hoitaa virtuaalikoneiden luonnin, pysäytyksen ja muun libvirt-kirjaston lävitse (Burden et al. 2014). Kuviosta 5 näkee VDSM:n toiminnan, ja se auttaa hieman konseptin hahmottamisessa. Kuvion 5 perusteella VDSM API:n ylitse voidaan hallita isäntäkoneen tilaa tekemällä koneelle muutoksia koneen peruskonfiguraatioon (levyt, verkot), jotka löytyvät kuvioista vasemmalta reunalta, tai käynnistää virtuaalikoneita libvirt-kirjaston ylitse.

3.7 SPM

Kun Ovirtia otetaan käyttöön, on pääkäyttäjän lisättävä datacenterille (oletuksena yksi luodaan valmiiksi, lisää voi luoda aina myöhemmin) hostien lisäksi tallennustila virtuaalikoneita, levykuvia ja snapshotteja varten. Kun tallennustila on lisätty datacenterille, pitävät hostit vaalit, kenestä muodostetaan SPM (Storage Pool Manager). SPM:n tehtävänä on kirjoittaa metadataa



Kuvio 5: VDSM arkkitehtuuri (alkup. kuvio Ovirt 2010)

tietovarastoon, ja se onkin ainut, joka näin voi tehdä. Muut hostit voivat suoraan lukea tallennustilaa, mutta kirjoittaminen kierrätetään aina SPM:n kautta.

Vikatilanteissa (SPM host vikaantuu) datacenterin hostit pitävät jälleen kerran vaalit ja manageri antaa SPM-tittelin toiselle hostille, jolla ei ole ongelmia.

4 Ovirt

4.1 Red Hat

Red Hat on amerikkalainen monikansallinen yritys, jonka tuotevalikoimaan kuuluu monia avoimen lähdekoodin ohjelmistoja, palveluita sekä konsultointia. Alunperin Ovirtia ei ollut laisinkaan tarjolla vapaasti, vaan Red Hat kehitti omaa virtualisointikonaisuutta, Red Hat Enterprise Virtualizationia (RHEV). Kun kehitystä oli suoritettu jonkin aikaa, päätettiin Red Hatilla julkaista RHEV avoimena lähdekoodina ja nimeksi päätettiin Ovirt. Samalla Ovirtista tehtiin upstream-julkaisu RHEVille. Red Hatilla on työntekijöitä yli 6100 ympäri

maailman. Red Hat on suurin yksittäinen organisaatio, joka kehittää Linuxia (Corbet, Kroah-Hartman ja McPherson 2012).

4.2 Ovirt

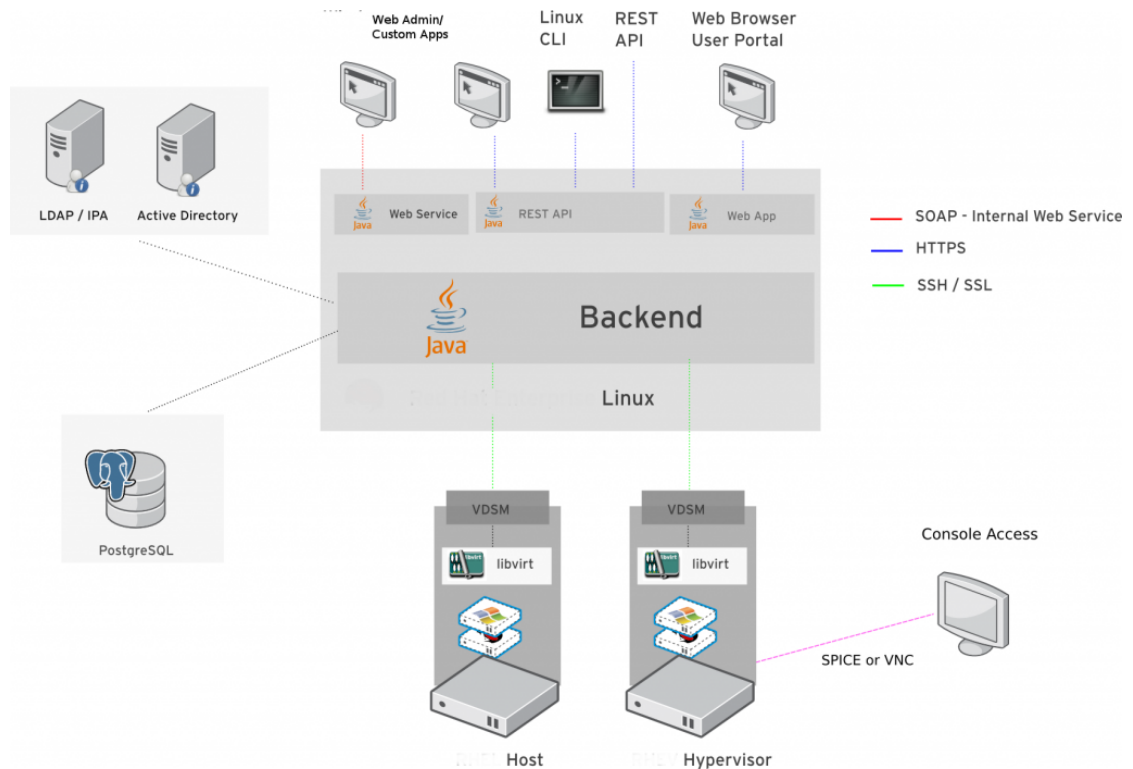
Ovirt on suhteellisen tuore avoimen lähdekoodin ohjelmistokokonaisuus, jonka taustalla on Red Hat. Red Hat tarjoaa myös omaa Red Hat Enterprise Virtualization (RHEV) -ohjelmistokokonaisuutta, jolle Ovirt on niin kutsuttu upstream-julkaisu. Upstream-julkaisuissa alkuperäinen kehittäjä lisää ominaisuuksia tuotteeseensa, mahdollisesti jopa muilta käyttäjiltä näin halutessaan. Koska Ovirt on upstream-julkaisu RHEV:lle, uudet ominaisuudet kehitetään ensin Ovirtille, kunnes Red Hat päättää ominaisuuksien olevan riittävän kypsiä ja ne integroidaan RHEV:iin.

Ovirt on virtualisointimanageri, jolla on tarkoitus nopeuttaa ja helpottaa kymmenien, satojen tai jopa tuhansien koneiden virtualisointia. Ovirt seuraa isäntäkoneiden tilaa, ja virhetilanteen tapahtuessa generoituu varoitus tai ilmoitus manageriin. Riippuen virhetilanteesta ja konfiguraatiosta pyrkii manageri siirtämään virtuaaliset koneet toisen hostin päälle. Managerin saa myös ilmoittamaan virhetilanteista sähköpostilla, tai raportoinnin voi yrittää integroida aikaisempaan käytössä olevaan työkaluun (mm. Nagiokselle pitäisi löytyä liitännäiset, mutta Ovirtin omien sivujen linkit liitännäisiin ovat epäkunnossa tai alkuperäinen kehittäjä on luopunut projektistaan).

Ovirt on rakennettu useasta avoimen lähdekoodin komponentista, jotka on yhdistetty Ovirt managerin avulla yhdeksi kokonaisuudeksi. Pääpiirteissään Ovirt rakentuu itse managerista, Libvirtistä, KVM:stä, QEMUsta sekä VDSM:stä.

Ovirtia voi verrata VMwaren vSphere-ohjelmistokokonaisuuteen tarjoten jotakuinkin samat toiminnot.

Työn kannalta Ovirt on täyden virtualisoinnin konsepti, jossa virtualisointi hoidetaan tyypin 1 hypervisoreilla, joita manageri ylläpitää asetettujen asetusten puitteissa (ks. kuvio 6). Tämän lisäksi Ovirt-manageri on myös mahdollista siirtää pilveen, jota se itse hallitsee.



Kuvio 6: Yleiskuva Ovirtistä (alkup. kuvio Ovirt 2010)

4.3 Ovirtin ominaisuuksia

Ovirt tarjoaa useamman eri toiminnon, joista keskeisimmät on listattu tähän. Tarjottuja ominaisuuksia myös testattiin käytännössä, ja ominaisuuksista onkin mainittuna lisää luvussa 7.

Korkea saatavuus (High Availability)

Jos isäntäkone syystä tai toisesta katoaa hallinnasta tai se vikaantuu, Ovirt käynnistää virtuaalikoneet automaattisesti toisilla koneilla, jos näin on niille määritetty.

Ovirt osaa myös siirtää vieraskoneet toisille hosteille niin, että käyttäjä ei huomaa virtuaalikoneen vaihdosta (live migration).

System Scheduler

Ovirt huolehtii asetettujen määritteiden perusteella virtuaalikoneet hostilta toiselle kuorman tasaamiseksi.

Virransäästö (Power Saving)

Asetettujen virransäästöominaisuuksien myötä Ovirtin saa määräämään virransäästöprofileja niin, että vain osa koneista on käynnissä. Määriteltyjen ehtojen mukaan koneet voivat käynnistyä ja sammua sopiviin aikoihin. Yksi

esimerkki tästä on, että koneita otetaan useampi käyttöön ruuhkatunteina ja sammutetaan, kun niitä ei tarvita. Tämä toiminne vaatii hostilta fyysistä tukea.

Huolto (Maintenance)

Hostit voidaan asettaa niin sanottuun huoltotilaan, jolloin asetettujen pohjaehtojen mukaan virtuaalikoneet siirtyvät muiden hostien huoleksi. Näin toimimalla saadaan isäntäkoneen päivitykset ohjelmistoihin tai fyysiseen laitteistoon tehdä rauhassa ilman käyttökatkoksia virtuaalikoneen käyttäjille.

Levykuvien hallinta

Virtuaalikoneista saa tehtyä niin kutsuttuja pohjavedoksia, joiden perusteella virtuaalikone asennetaan käyttövalmiiksi. Pohjavedokset nopeuttavat virtuaalikoneen käyttöönottoa. Esimerkiksi tietyn ryhmän käyttöön tarkoitetuilla virtuaalikoneilla on kaikilla samat asetukset, kunnes käyttäjä muokkaa niitä. Lisäksi levykuvien hallinnalla voidaan luoda käynnissä olevista koneista snapshotteja. Snapshotit tallentavat koneen sen hetkisen tilan, ja esimerkiksi ohjelmistopäivitysten yhteydessä peruuttaminen aiempaan snapshottiin palauttaa tilan siihen, mitä se oli ennen ohjelmistopäivityksiä. Riippuen käytetystä virtuaalikoneen levytyypistä (thin tai thick) snapshotin tekeminen vie muutaman sekunnin tai useamman minuutin.

Seuranta & raportointi

Ovirt seuraa virtuaalikoneiden tilaa ja resurssien käyttöä. Resurssien käyttö tallennetaan logitiedostoihin ja näistä voidaan generoida automaattinen raportti.

OVF tuonti/vienti

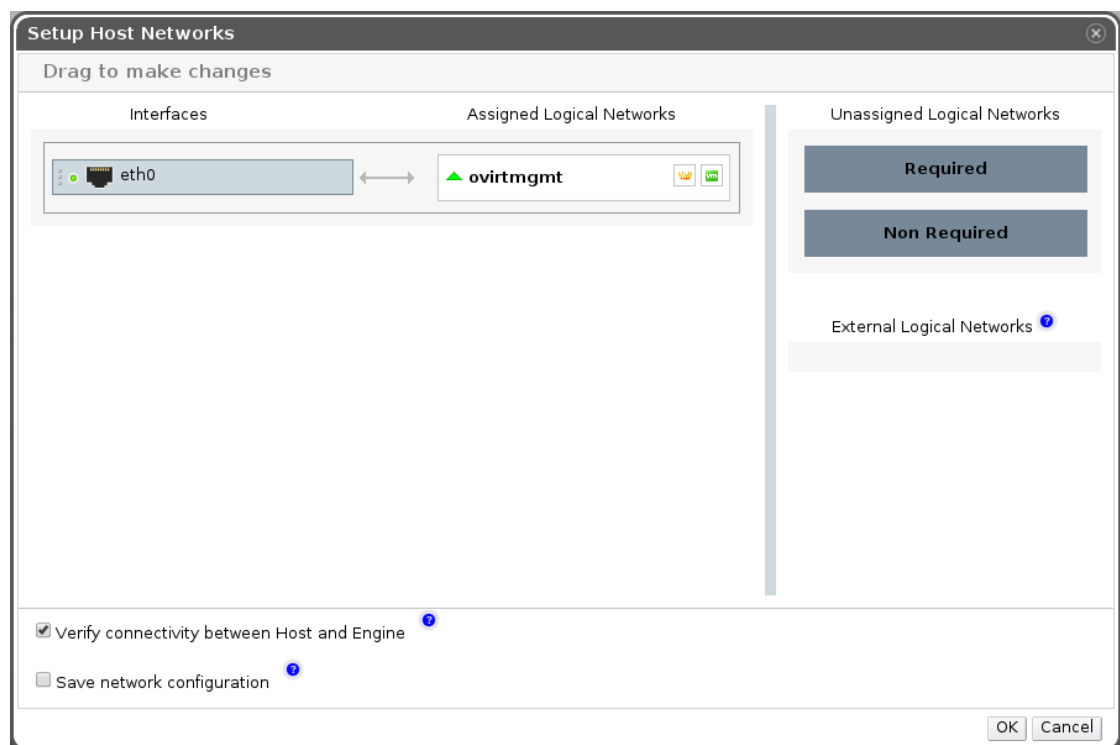
Virtuaalikoneita voi Ovirtille tuoda ja viedä Open Virtualization Format-muodossa muista virtualisointialustoista. OVF on avoin standardi, ja ainakin osa virtualisointityökaluista on ottanut OVF:n käyttöönsä.

Verkot

Ovirtissä on mahdollista luoda verkkorajapintoja virtuaalikoneiden käytettäväksi ja jopa erotella liikennettä tietyin määrityksin. Hostien fyysiset rajapinnat on mahdollista yhdistää yhdeksi sovittimeksi, jolloin koneilta lähtevä ja saapuva liikenne saadaan yhdistettyä yhteen loogiseen verkkorajapintaan. Kuviossa 7 on nähtävillä verkkorajapinnan manipulointi.

Käyttäjätilit

Ovirtiin itseensä ei ole mahdollista lisätä paikallisia käyttäjätilejä ainakaan vielä, mutta apuna käytettävän autentikointipalvelimen kanssa voi Ovirtiin lisätä käyttäjille erinäisiä oikeuksia. Tässä työssä apuvälineenä autentikointia varten oli FreeIPA, josta hieman lisää luvussa 6. Tärkeimmät käyttäjäryhmät ovatkin pääkäyttäjä (admin), muut pääkäyttäjät (verkolle, tallennustilalle jne.), tehokäyttäjät (power users) ja normaalit käyttäjät (users). Pääkäyttäjän tunnuksilla on mahdollisuus toki hallita kaikkea, mitä Ovirt-pilveen on tuotu. Muille pääkäyttäjille on mahdollista antaa jotkin osa-alueet Ovirt-pilvestä, jota he voivat hallinnoida, mutta muihin osa-alueisiin he eivät voi tehdä muutoksia. Tehokäyttäjille annetaan oikeudet hallinnoida pilven päälle pystytettyjä virtuaalikoneita, luoda niitä tai lisätä valmispohjia normaaleille käyttäjille. Peruskäyttäjillä on luonnollisesti vähiten oikeuksia, ja he voivatkin tehdä virtuaalikoneilleen perusoperaatioita kuten käynnistyksen ja sammutuksen. Toki virtuaalikoneen sisällä heillä voi olla mitkä oikeudet tahansa.



Kuvio 7: Verkkorajapinnan manipulointi Ovirtin hallinnassa

5 Levyjärjestelmät

Ovirtin dokumentaation mukaan ohjelmisto tukee useampaa tallennusratkaisua palvelun käyttöön. Riippumatta käytetystä tallennustilan ratkaisusta, tilan liittämisen jälkeen Ovirtille lisätyt isäntäkoneet (hostit) järjestävät vaalit. Vaaleissa koneet äänestävät, mistä isäntäkoneesta tulee SPM (Storage Pool Manager). SPM:n tehtävänä on kirjoittaa metadata kaikkien muiden Ovirt-isäntäkoneiden puolesta levyille. Näin välttään tilanteesta, jossa metadataa tallennettaessa koneet tallentaisivat toistensa kanssa yhtäaikaa samoja tietoja tai jopa toistensa tietojen päälle. Tämä toki aiheuttaa myös ongelmia, koska SPM:n kaatuessa myös muut hostit lakkaavat toimimasta, kunnes uudet SPM vaalit on pidetty ja korvaaja vikaantuneelle SPM:lle on valittu.

Levyjärjestelmissä on hieman päällekkäisyyksiä, koska esimerkiksi Glusterin tapauksessa voidaan levyjako tehdä NFS tai iSCSI tyyppisesti.

Koska vaihtoehtoja on runsaasti saatavilla, on tallennusratkaisun valinta hieman haasteellista Ovirtille. Valintaa onkin syytä punnita hetki uuden Ovirt-pilven kanssa.

5.1 iSCSI

iSCSI (Internet small computer system interface) luotiin siirtämään SCSI-komentoja IP-verkon yllä. Koska SCSI-komennot kulkevat verkon päällä ja ne puretaan päissä, saadaan asiakaskäyttöjärjestelmät uskoteltua siihen, että ne käyttäisivät paikallista levyä.

iSCSI-jaot toimivat hieman muista ratkaisuista eri tavalla. iSCSI:n tapauksessa tallennustila jaetaan palvelimelle määritetyillä levyillä kokonaisuudessaan niin, että siihen liitetty tietokone ei tiedä levyn olevan verkossa.

iSCSI:n tapauksessa palvelimesta käytetään nimitystä target ja verkon yli levyä käyttävästä koneesta käytetään nimitystä initiator. Initiator pyytää palvelimelta levyä käyttöön ja jos tätä ei ole estetty millään tavalla, saa initiator levyn käyttöönsä. Levy on hieman harhaanjohtava termi, sillä levy ei välttämättä ole yksi fyysinen komponentti, vaan se voi koostua useista eri kiintolevyistä useilla eri palvelimilla. Tallennustilasta käytetäänkin termiä LUN (Logical Unit

Number). Tallennustilaa ei ole järkevää käyttää niin, että siinä olisi kiinni useampi eri initiator, koska koneet voivat kirjoittaa toistensa tietojen päälle ellei levy(pakan) tiedostojärjestelmä ole suunniteltu tätä varten (mm. Red Hatin GFS2 tai VMwaren VMFS).

5.2 GlusterFS

GlusterFS tai lyhyemmin pelkkä Gluster on hajautettu levyjärjestelmä. Tämä tarkoittaa sitä, että levyt ovat useammilla eri tietokoneilla ja datan replikointi voidaan toteuttaa useammalla eri politiikalla. Data voi olla kahdennettua, niin että samat tiedot löytyvät useammalta Gluster-koneelta tai se voi olla vain hajautettuna tai näiden yhdistelmä.

Jos työn alkuvaiheessa olisi Glusteria tutkittu hieman enemmän, olisi levynä Ovirt-pilvelle käytetty Glusteria. Tämä vain sillä pohjalla, että Red Hat on Glusterin takana kehittämässä ratkaisua ja integroinnin osalta Ovirtin kanssa se varmasti sopisi viimeistään muutaman version päästä, jos ei jo nyt, erittäin hyvin. Lisäksi tietyillä ominaisuuksilla Ovirtistä saisi ehkä hieman vikasietoisemman, koska data pilvestä olisi mahdollisesti kahdennettu kahden palvelimen välillä.

5.3 NFS

NFS (network file share) on tiedostopohjainen jakotapa. Erona aiemmin mainittuun iSCSI-levytyyppiin NFS-asema voidaan lisätä useammalle eri koneelle yhtäaikaaisesti ilman ongelmia. Levy otetaan käyttöön siihen liitetyllä koneella ja kone jakaa (ja antaa kirjoittaa) tiedostoja muilta koneilta.

NFS-jakojen tapauksessa voi levyn jakaa useamman eri tietokoneen kanssa ilman pelkoa, että toinen kone ylikirjoittaisi vahingossa dataa, koska NFS-palvelin vastaanottaa paketit ja kirjoittaa itse tiedon haluamallaan tavalla levyille.

Erityisesti iSCSI:in verrattuna NFS-kone vaatii roimasti suurempaa suorituskykyä johtuen levyn jakotavasta, mutta on oletuksena myös hieman turvallisempi useilla levyille kirjoittavilla koneilla.

5.4 Muut levytila ratkaisut

Ovirtin kanssa on mahdollista ottaa käyttöön myös paikallinen levy, mutta tämänkaltainen ratkaisu ei ole järkevä tuotantoympäristössä, koska ratkaisu ei skaalaudu. Lisäksi kyseisen hostin vikaantuessa on levyjärjestelmä täysin muiden hostien saavuttamattomissa. Pikaiseen testailuun tämä on varmasti omiaan, koska infran pystyttämiseen voi käyttää vähemmän aikaa ja itse tuotteen arviointiin pääsee huomattavasti nopeammin. Lisäksi jos levyjärjestelmä on POSIX-yhteensopiva, voi sen lisätä Ovirtiin, mutta jako on tehtävä NFS-tyyppisesti.

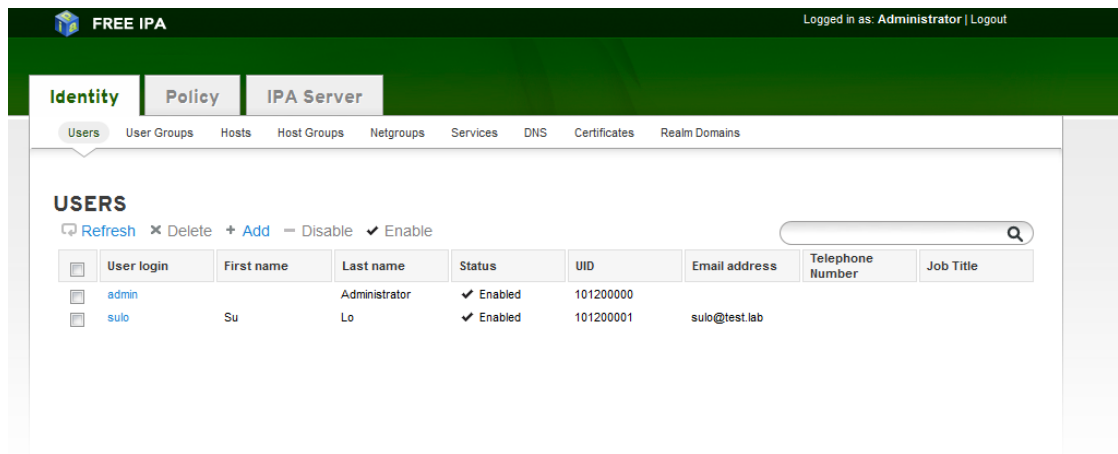
6 FreeIPA

FreeIPA on avoimen lähdekoodin identiteetti-, käytänne- (policy) ja auditointipalvelin, joka on pääasiassa suunnattu Linux- ja Unix-käyttöjärjestelmiä varten. FreeIPA koostuu 389 Directory Serveristä (Lightweight Directory Access Protocol, LDAP) ja MIT:n (Massachusetts Institute of Technology) kehittämästä Kerberos-todennusprotokollasta. Lisäksi FreeIPAan on lisätty aikapalvelin (NTP), nimipalvelin (DNS) ja sertifikaattien hallinta (Dogtag). Tämänkin projektin taustalla on Red Hat sponsoroimassa projektia. FreeIPAA voi verrata Microsoftin Active Directoryyn. Kuviosta 8 pääsee tutustumaan FreeIPAn päänäkymään.

Palvelimelle voidaan luoda käyttäjätilejä, määritellä niitä ryhmiin ja antaa oikeuksia ja käytänteitä. Näin toimimalla pystytään käyttäjille tarjottavia palveluita sallimaan, rajoittamaan tai estämään halutuin määrityksin. Työn kannalta olisi ollut järkevää käyttää FreeIPAA nimipalvelimena Ovirt-alustalle, koska se siihen on integroitu, mutta FreeIPA-palvelin otettiin vasta myöhemmin käyttöön työn toteutuksen aikana.

7 Käytäntö

Tilaajan toimesta käyttöön annettiin kaksi tietokonetta. Molemmat tietokoneista olivat seuraavanlaisia:

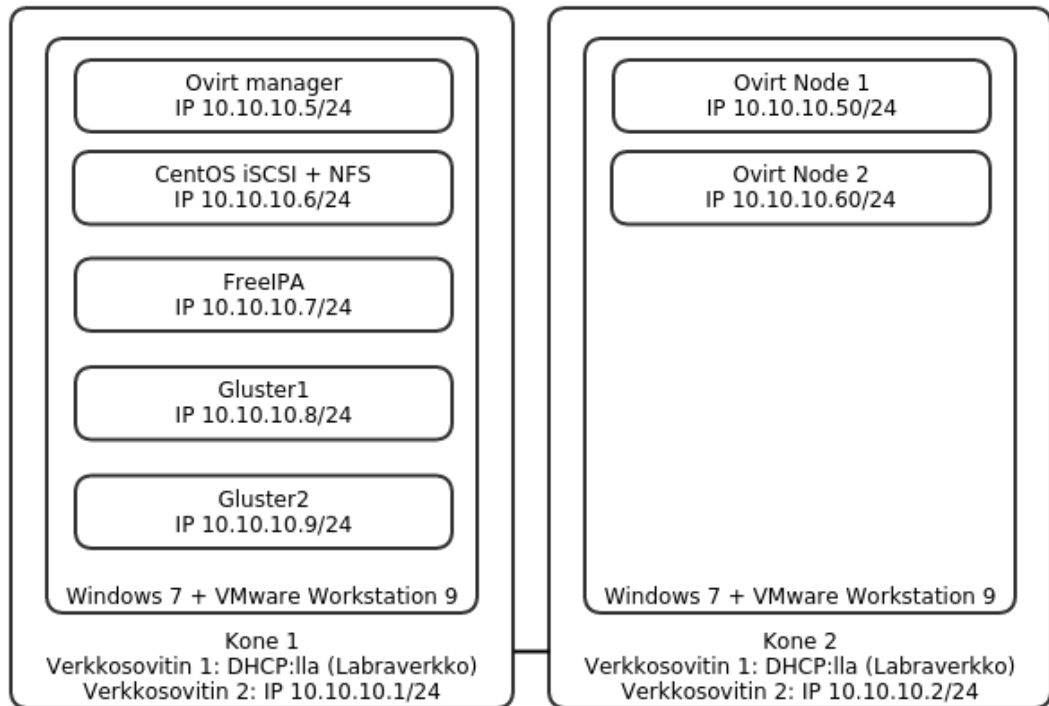


Kuvio 8: FreeIPAn päänäköymä Admin käyttäjälle

- Suoritin: Intel Core i5 750 (2.67GHz)
- Keskusmuisti: 8Gt
- Kiintolevytila: 500Gt
- Verkkoyhteydet: kaksi kappaletta gigabitin verkkosovittimia
- Käyttöjärjestelmä: Windows 7
- Virtualisointi: VMware Workstation 9

Koneille virtualisoitiin työn kannalta tarvittavat virtuaalikoneet kuvion 9 mukaisesti. Kuten kuviosta voi päätellä, koneen kaksi Ovirt-isäntäkoneet hoitavat suurimman työn, joten näin roolit jakamalla saatiin laiteresurssit optimoitua hyvin. Verkkosovittimien osalta molemmat koneista olivat kytkettyjä Jyväskylän ammattikorkeakoulun labraverkkoon suoraan ja toisten sovittimien osalta koneet olivat toistensa kanssa ristiinkytettyjä.

Koska käytössä oli vain kaksi fyysistä konetta ja suunnitelmissa oli testata live migration ja muita toimintoja, piti työssä virtualisoida useampia koneita fyysisen laitteiston päälle. Koneilla oli valmiina asennettuna tilaaajan puolesta Windows 7-käyttöjärjestelmät, joille oli asennettu VMware Workstation -virtualisointiohjelmistot ja ammattikorkeakoulun tarjoama lisenssi ohjelmistolle. Näin toimimalla päästiin tilanteeseen, jossa rajoitetulla laitemäärällä saatiin riittävä määrä virtuaalisia tietokoneita käyttöön. Virtuaalisten koneiden päälle oli tarkoitus asentaa Ovirt, jolla oli tarkoitus vielä testata Ovirt-virtualisointia. Virtualisoidut koneet pidettiin omassa virtuaaliverkossaan, että labraverkolle ei



Kuvio 9: Virtuaalikoneiden jakautuminen fyysisille koneille sekä näiden IP-osoitteistus.

aiheutuisi ongelmatilanteita ja että labraverkon asetukset eivät vaikuttaisi Ovirt-koneisiin.

Yhdelle virtuaalikoneista asennettiin Ovirt-manageri, yhdelle asennettiin tallennustila pilvelle iSCSI ratkaisuna, yhdelle asennettiin FreeIPA ja kahdesta virtuaalikoneesta tehtiin isäntäkoneita Ovirtin hallinnoimille virtuaalikoneille. Manageri asennettiin Fedora 19:n päälle, iSCSI-jaot tehtiin Centos 6.5:llä ja isäntäkoneista tehtiin Ovirtin sivulta löytyvän levykuvan perusteella Ovirt-isäntäkoneita (host). Isäntäkoneille asennettiin vielä virtuaalikoneet, jotta live migrationin ja muiden ominaisuuksien toteutusta pystyisi seuraamaan.

Verkkoasetukset pidettiin erittäin yksinkertaisina ja VMware Workstationiin tehtiin yksi NATattu verkko (Network Address Translation) 10.10.10.0/24, joka sidottiin kummankin fyysisen koneen ristiinkytkettyyn verkkosovittimeen.

Verkolla oli lupa käyttää Windows-koneiden toista verkkorajapintaa, jos tarvetta internetille oli.

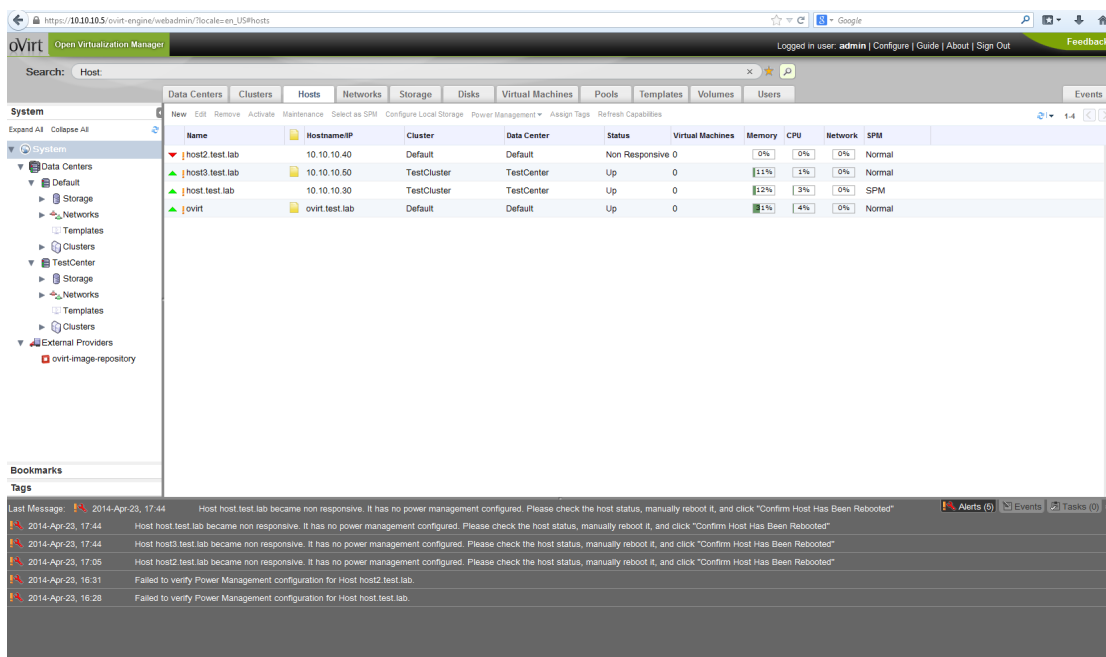
7.1 Manageri

Managerin asennus aloitettiin katsomalla vaatimuslistaa Ovirtin dokumentaatiosta. Minimiksi koneen resursseille on annettu kaksitytiminen suoritin, 4 gigatavua keskusmuistia, 25 gigatavua kiintolevytilaa ja gigabitin verkkosovitin. Suositelluissa resurssivaatimuksissa oli kaksi-/nelitytiminen suoritin, 16 gigatavua muistia, 50 gigatavua tallennustilaa useilla eri levyillä ja gigabitin verkkosovitin.

Suositusvaatimuksia tutkailemalla tuli todettua, että tilaajan toimittamat koneet eivät tule täyttämään näitä ainakaan muistin osalta. Tämä ei kuitenkaan asennusta estänyt, vaan Workstationiin luotiin virtuaalinen kone ylläolevilla minimivaatimuksilla. Virtuaalikoneen käyttöjärjestelmäksi asennettiin Fedora 19 suositusten mukaan. Alun pikaisen kokeiluvaiheen aikana tuli testattua myös uudempaa Fedora 20 levykuvaa, mutta tälle management ei suostunut asentumaan vaan törmäsi useisiin pakettiristiriitoihin.

Asennus aloitettiin määrittämällä VMware Workstationiin sisäinen verkko ja sitomalla se fyysisten koneiden välillä olevalle linkille. Kun verkko oli luotu, asennettiin Fedora 19 minimiasennuksena (ainoastaan peruskomponentit). Asennuksen päätteeksi koneelle lisättiin Ovirtin pakettivarastot, muokattiin koneen hosts-tiedostoa DNS-palvelimen (domain name server) puuttuessa ja asennettiin management-engine. Asennus ei sujunut täysin ongelmitta, vaan asennus tökkäsi pakettiristiriitaan. Koska tästä ei ollut mainintaa dokumentaatiossa, jouduttiin virheilmoitus kaivamaan lokista ja tarkistamaan internetistä syytä. Ongelmalle löytyi ratkaisu, joka johtui varusohjelmien päivittämättömyydestä pakettivarastoihin ja asennusta päästiin jatkamaan. Tästä syystä liitteessä 9 on hieman eroavaisuutta Ovirtin omaan dokumentaatioon. Management saatiin toimintakuntoon ja koneen palomuurista avattiin portit ulkopuolelle hallinnan helpottamiseksi.

Ovirtin päänäkyvä asennuksen päätteeksi ja muutaman hostin ja levytilan lisäämisen jälkeen (kuvio 10).



Kuvio 10: Ovirtin hallintapaneeli hostien ja tallenustilan lisäyksen jälkeen

7.2 Levytila

Ovirt-pilvelle välttämätön levytila työn osalta tehtiin Centos 6.5 GNU/Linux julkaisun päälle. Centos-kone asennettiin Centosin sivuilta ladatulla levykuvalla, mutta tässä vaiheessa VMware Workstationin automatiikan vuoksi koneelle tuli asennettua palvelinasennukseksi liikaa tavaraa. Koska virtuaalikoneen asetuksissa oli koneelle määritetty, että käyttöjärjestelmäksi asennetaan Centos, valitsi automatiikka (asennusskriptit), että koneelle asennetaan graafinen työpöytä. Tämä ei sinällään haitannut, mutta koneen tarvitsemia resursseja olisi voinut pudottaa hiukan ilman työpöytämanageria. Minimivaatimuksia ei juurikaan tämänkaltaisessa levyjaossa ole, koska iSCSI:n tarjoama levytila kuluttaa isäntäkoneen resursseja verkkoa lukuunottamatta hyvin vähän.

Kun asennus oli valmistunut, lisättiin pakettivarastoista iSCSI-target, liitettiin koneelle pari kiintolevyä ja määritettiin kyseiset levyt jaettaviksi koko verkolle ilman käyttäjätunnuksia. Asennuksesta tarkemmin liitteessä 9. Kuvio 11 on nähtävillä jaettu levy (backing store) sekä sallitut hostit (initiator address). Tuotannossa tätä ei kannata tehdä näin, koska tietoturvariski on aina olemassa. Lisäksi palomuriin sallittiin jaot koko verkon alueelta (kuvion 12 rivit joissa –dport 3260) ja käynnistettiin palvelu.

```
<target iqn.2014-04.lab.test:storage.centos>
backing-store /dev/sdb1
initiator-address 10.10.10.0/24
</target>
[root@localhost ovirt]#
```

Kuvio 11: Centos palvelimen iSCSI jako

```
[root@localhost ovirt]# cat /etc/sysconfig/iptables
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 3260 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 3260 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
[root@localhost ovirt]#
```

Kuvio 12: Centos iSCSI palvelimen IPtables säännöt

7.3 Isäntäkoneet

Isäntäkoneiden osalta laitevaatimukset ovat kaikkein suurimmat, koska nämä koneet tekevät käytännössä kaiken työn virtuaalikoneiden osalta. Mitä enemmän virtuaalikoneita isäntäkonetta kohden asennetaan, sitä enemmän koneelta vaaditaan erityisesti muistia. Joissakin tapauksissa myös suorittimen suorituskyky voi jäädä pullonkaulaksi. Minimivaatimukset Ovirtin oman dokumentaation mukaan on kahden ytimen suoritin, jossa on joko Intel VT tai AMD-V -virtualisointilaajennokset, 10 gigatavua muistia, 10 gigatavua kiintolevytilaa ja gigabitin verkkosovitin. Suosituksina dokumentaation osalta pidetään kahta suoritinta (eri asia kuin kahden ytimen suoritin), jossa on samat laajennokset kuin yllä, 16 gigatavua muistia, 50 gigatavua kiintolevytilaa ja gigabitin verkkosovitin. Kuten vaatimuslistaa tavaamalla voidaan todeta, ei tilaajan toimittamien fyysisten koneiden muistimäärä riitä edes minimiin, mutta asennusohjelma ei asiasta huomauttanut tai valittanut.

Hostien asentaminen oli tämän projektin vähiten toimenpiteitä vaativa suoritus, kun ongelmakohdat oli selvitetty. Asennus ei aluksi suostunut menemään läpi,

vaan juuttui hieman asennuksen aloittamisen jälkeen. Pienen vianselvityksen jälkeen asennus onnistui ja isäntäkoneet saatiin toimintakuntoon. Tästä aiheesta enemmän luvussa 9. VMware Workstationiin määritettiin kaksi virtuaalikonetta, joista molemmille annettiin puolet koneen muistista (4 gigatavua), tuplaydinsuorittimet ja hieman levytilaa. Lisäksi VMware workstationista tuli lisätä Intel VT / AMD-V suoritinlaajennoksiin merkintä, että virtuaalikoneet voivat jakaa laajennosta omille virtuaalikoneilleen. Levykuva ladattiin Ovirtin sivuilta ja asennus aloitettiin muutamalla lisäoptiolla, jotka puuttuivat dokumentaatiosta. Tällä kohdalla ei dokumentaatiosta löytynyt kohtaa, miksi asennus juuttuu, mutta kernelin lisäoptioita käynnistysvaiheessa tutkimalla päätettiin poistaa hiljainen asennus, jotta ongelma selviäisi. Asennus saatiinkin onnistumaan ja hostien lisääminen managementtiin oli muutaman option syöttäminen koneen käynnistyttyä. Yksi mahdollinen selitys asennuksen epäonnistumiselle oli pääkäyttäjän eli rootin salasanan puute, sillä tätä tuli testattua asennuksen aikana. Liitteessä 9 mainituilla lisäoptioilla koneen sai käyntiin ja asennuksen alkamaan, mutta ilman pääkäyttäjän salasanaa asennus juuttui mystisesti 20 prosentin tietämille. Täyttä varmuutta ei asiasta kuitenkaan ole, koska logitietoihin ei päässyt käsiksi asennuksen hyydyttyä.

7.4 FreeIPA

Tilaaajan puolesta haluttiin selvittää myös, miten käyttäjäkohtaiset tilit toimivat Ovirtin kanssa. Yksi mahdollisuus olisi ollut yrittää liittää Jyväskylän ammattikorkeakoulun tuotantoverkon Windows Active Directory -koneet Ovirtin AAA-palvelimiksi (Authentication, Authorization ja Accounting), mutta virheellisten määritysten ilmetessä voisi tuotantoverkolle tapahtua jotain ikävää. Vaikka tilaaja oli antanut varsin roimat oikeudet työhön käytettäville koneille, olisi tämä pyyntö mitä todennäköisimmin evätty. Tästä syystä otettiin käyttöön hallinta- ja levytilakoneelle yksi ylimääräinen virtuaalikone alkuperäisen suunnitelman lisäksi, johon asennettiin Fedora Core 19 minimiasennuksena. Kyseiselle palvelimelle asennettiin FreeIPA-paketit. Palvelin asetettiin keksityn toimialueen (test.lab) AAA-palvelimeksi ja palvelimelle lisättiin muutama käyttäjä, joilla oli erilaisia oikeuksia käyttää ja ylläpitää tietyn ryppään

virtuaalikoneita Ovirt-pilvestä. Tämän palvelimen asennus pitäisi onnistua liitteen 9 käskyillä.

7.5 Kokonaisuus

Virtuaalikoneiden (manageri, hostit, FreeIPA ja levypalvelin) asennuksen päätteeksi aloitettiin konfiguroimaan Ovirtia. Konfigurointi alkaa aina kirjautumalla web-hallintaan management-koneelle, jonka IP-osoite määritetään asennusvaiheessa. Management koneella voi entuudestaan olla HTTP-palvelin (hyper text transfer protocol) ja managementin saa sijoitettua palvelimen juuren sijaan johonkin alikansioon, mutta oletusasetuksilla asennus muokkaa koneen asetuksia ja asettaa itsensä HTTP-palvelimen juureen uudelleenohjauksilla. Kun palveluun kirjaudutaan, ei sieltä välttämättä löydy mitään muuta kuin oletusdatakeskus.

Ensimmäiseksi luotiin uusi datakeskus sopivilla määrittäyksillä (yhteensopivuus version osalta, levytyyppi), muokattiin asetuksia isäntäkoneilla ja hyväksyttiin ne hallinnassa haluttuun datakeskukseen. Isäntäkoneiden datakeskukseen lisäyksen jälkeen voi kyseiselle datakeskukselle lisätä levypalvelimen. Työssä tehdylle TestCenter-datakeskukselle lisättiinkin iSCSI-palvelin graafisesti toimimalla hallinnasta löytyvän guide me -avustajan avulla. Avustaja pyytää halutun levyn tiedot (IP-osoite tai hostname), tyyppin (iSCSI, Gluster, NFS) ja Ovirt-isäntäkoneen, joka ottaa ensimmäisen SPM (storage pool manager) vuoron. Kun levytila on lisätty, suostuvat isäntäkoneet vaihtamaan tilansa down-tilasta tilaan up (katso kuvion 10 status saraketta) ja isäntäkoneille voidaan alkaa lisäämään virtuaalikoneita. Työssä lisättiin datakeskukselle myös vielä ISO-domain NFS-tyyppisesti, jonne alustavat levykuvat virtuaalikoneille saa lisättyä sekoittamasta varsinaista ja muokattavaa dataa iSCSI-levyltä. ISO-domainiin ladattiin levykuva (Fedora 20) ja asennettiin se käytettäväksi virtuaalikoneeksi. Kyseistä Fedoraa käytettiin testaamaan live migration.

Live migration -testi toimi muuten hyvin, mutta hallintayhteys virtuaalikoneen työpöytänäkymään katkeaa, kun virtuaalikone siirtyy hostilta toiselle. Kuten kuvio 13 näkyy, aloittaa Ovirt virtuaalikoneen siirtämisen toiselle koneelle, mutta siirron aikana Virt-Viewer-ohjelma lakkaa näyttämästä koneen hallintaa.

Tämän aiheuttaa Ovirtin ominaisuus, jossa koneen työpöytänäkymään luodulle yhteydelle tehdään kertakäyttöinen avain. Siirron aikana avain vanhenee, koska se on isäntäkone perustainen ja varmistettu myös oletuksena 180 sekunnin vanhentumisajalla. Koneelle saa yhteyden välittömästi siirron valmistuttua eikä koneella tapahdu yhteyskatkoksia. Verkkoyhteyden katkeamattomuus voidaan todeta ping-ohjelman avulla. Pingatessa toista isäntäkoneista, ei paketteja huku yhtään, mutta live migration aiheuttaa hienoisen nousun viiveessä siirron aikana. Testiympäristössä virtuaalikone ei siirtynyt fyysiseltä koneelta toiselle lainkaan työssä käytetystä suunnitelmasta johtuen. Kuviossa 14 näkyy havainnollistava esimerkki viiveen noususta. Graafisen hallintayhteyden voi koneeseen muodostaa myös selaimella, mutta ominaisuus vaatii selainlaajennoksia. Laajennoksia ei ollut saatavilla Windows-käyttöjärjestelmille käytännön osuuden aikana.

Last Message:	✓ 2014-May-22, 13:41	Migration started (VM: Fedora-20, Source: host.test.lab, Destination: host3.test.lab, User: admin).
✓ 2014-May-22, 13:41		Migration started (VM: Fedora-20, Source: host.test.lab, Destination: host3.test.lab, User: admin).
✓ 2014-May-22, 13:37		User admin@internal is connected to VM Fedora-20.
✓ 2014-May-22, 13:37		user admin initiated console session for VM Fedora-20
✓ 2014-May-22, 13:37		VM Fedora-20 started on Host host.test.lab
✓ 2014-May-22, 13:36		Storage Pool Manager runs on Host host.test.lab (Address: 10.10.10.30).
✓ 2014-May-22, 13:36		Reconstruct Master Domain for Data Center TestCenter completed.
✓ 2014-May-22, 13:36		VM Fedora-20 was started by admin (Host: host.test.lab).

Kuvio 13: Live migration siirto hostilta toiselle

```

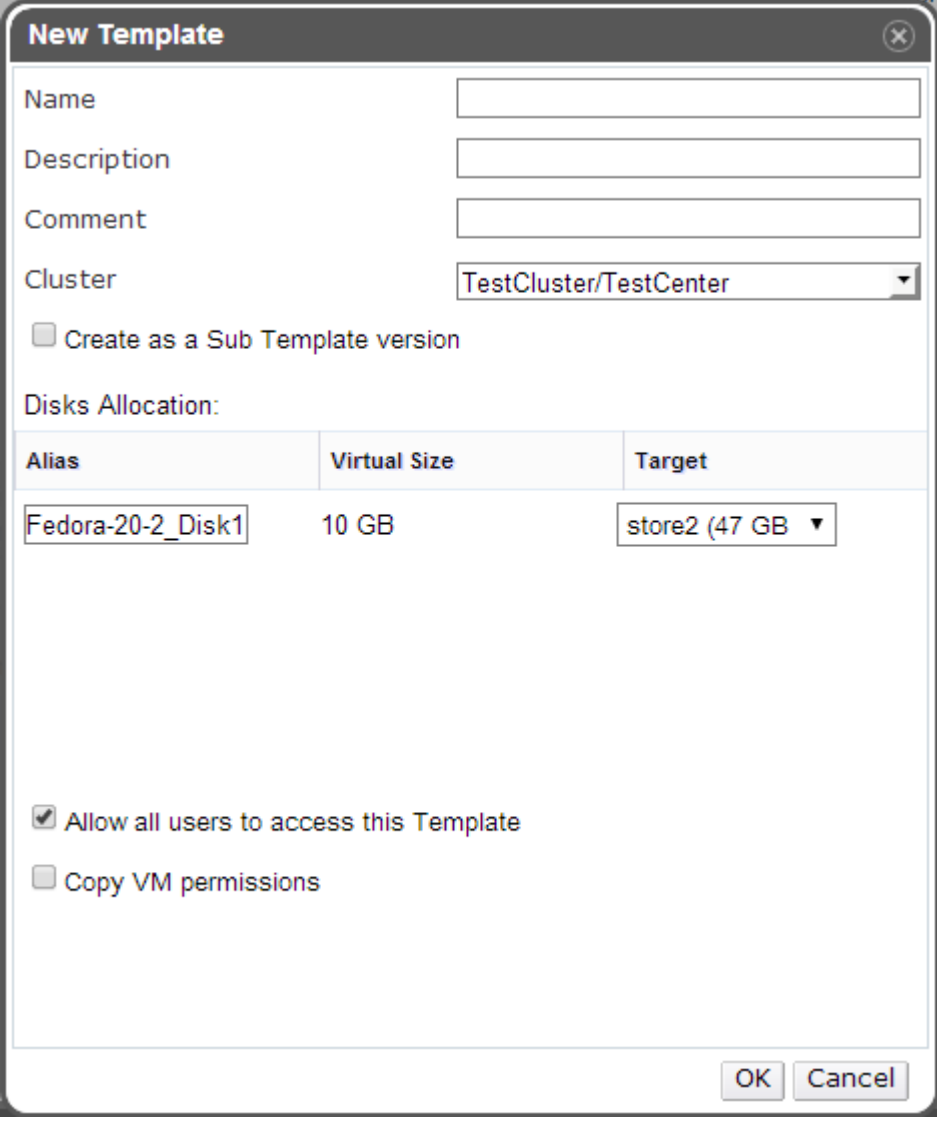
64 bytes from 10.10.10.30: icmp_seq=47 ttl=64 time=1.10 ms
64 bytes from 10.10.10.30: icmp_seq=48 ttl=64 time=1.11 ms
64 bytes from 10.10.10.30: icmp_seq=49 ttl=64 time=1.12 ms
64 bytes from 10.10.10.30: icmp_seq=50 ttl=64 time=1.18 ms
64 bytes from 10.10.10.30: icmp_seq=51 ttl=64 time=1.71 ms
64 bytes from 10.10.10.30: icmp_seq=52 ttl=64 time=2.38 ms
64 bytes from 10.10.10.30: icmp_seq=53 ttl=64 time=0.795 ms
64 bytes from 10.10.10.30: icmp_seq=54 ttl=64 time=0.891 ms
64 bytes from 10.10.10.30: icmp_seq=55 ttl=64 time=1.21 ms
64 bytes from 10.10.10.30: icmp_seq=56 ttl=64 time=1.02 ms
64 bytes from 10.10.10.30: icmp_seq=57 ttl=64 time=1.00 ms
64 bytes from 10.10.10.30: icmp_seq=58 ttl=64 time=1.18 ms
64 bytes from 10.10.10.30: icmp_seq=59 ttl=64 time=1.04 ms
64 bytes from 10.10.10.30: icmp_seq=60 ttl=64 time=0.880 ms
64 bytes from 10.10.10.30: icmp_seq=61 ttl=64 time=0.984 ms

```

Kuvio 14: Pingin hyvin pieni nousu siirron aikana

Ovirt-pilvessä tehtiin myös virtuaalikoneista valmiita pohjia (template), jotta virtuaalikoneen saisi nopeasti käyttöönsä haluaamaansa tarkoitukseen. Kuviossa 15 on näkyvillä pohjavedoksen luontiin tarkoitettu lomake.

Työn edetessä haluttiin myös testata minkälaisia mahdollisuuksia ja rajoitteita erilaisilla käyttäjätileillä pystyisi tekemään. Tätä varten FreeIPAan luotiin



New Template [X]

Name

Description

Comment

Cluster

☐ Create as a Sub Template version

Disks Allocation:

Alias	Virtual Size	Target
<input type="text" value="Fedora-20-2_Disk1"/>	10 GB	<input type="text" value="store2 (47 GB)"/>

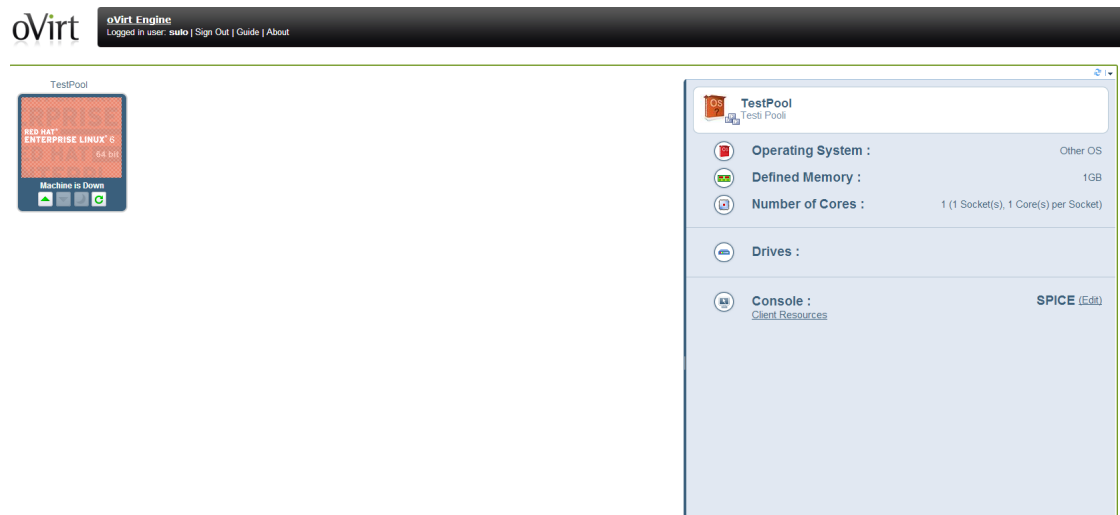
☒ Allow all users to access this Template

☐ Copy VM permissions

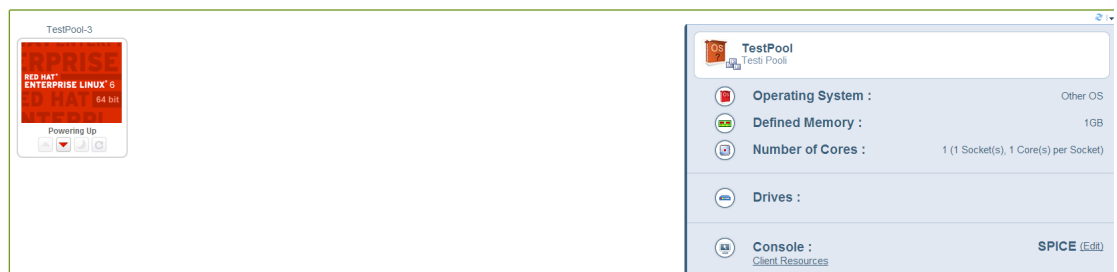
OK Cancel

Kuvio 15: Templaten luonti Ovirtissä

käyttäjii ja Ovirt-pilvestä annettiin käyttäjille eri oikeuksia virtuaalikoneiden hallintaan. Pääkäyttäjätillä oli luonnollisesti kaikki oikeudet ja pääkäyttäjän lisäksi luotiinkin käyttäjä Sulo sekä Salo. Sulolle annettiin oikeudet käynnistää, sammuttaa ja luonnollisesti myös käyttää datakeskuksen virtuaalikoneita. Salolle annettiin tehokäyttäjän oikeudet. Kuviosta 16 voi nähdä, kuinka erilainen näkymä on pelkälle virtuaalikoneiden käyttäjälle verrattuna pääkäyttäjän näkymään (kuvio 10). Kuviossa 17 virtuaalikone on käynnistymässä.



Kuvio 16: Normaalikäyttäjän näkymä kirjautuessa



Kuvio 17: Virtuaalikoneen käynnistys peruskäyttäjällä

Työssä haluttiin myös saada monitorointia Ovirt-pilven tilaa varten. Monitorointia varten Ovirtille on tarjolla liitännäiset, joilla pitäisi pystyä tarkastelemaan koneita Ovirt asennuksesta. Monitorointia yritettiin saada aikaan asentamalla yksi ylimääräinen virtuaalikone, johon asennettiin Nagios. Nagiokselle olisi pitänyt olla tarjolla liitännäiset Ovirtin monitorointiin, mutta Ovirtin sivujen tarjoamat linkit olivat epäkunnossa. Hallintaan olisikin pitänyt tulla oletusasennuksen päälle muutama ylimääräinen toiminto, mutta valitettavasti Ovirtin omien sivujen tarjoamat linkit ja ohjeet Nagios-liitännäisiin eivät toimineet. Ohjeiden ja latausten puuttuessa liitännäisiä lähdettiin

hakemaan internetistä. Liitännäiset löydettiinkin Github-palvelusta, mutta liitännäisten latausten yhteyteen lisättyjen suppeiden asennusohjeiden ollessa suorastaan surkeat ja tämän päälle työn loppuvaiheesta aiheuttama kiire alkoi painamaan päälle, ei integraatiota saatu tehtyä laisinkaan.

8 Vertailua vSphereen

Ovirtia haluttiin myös hieman vertailla sen suoraan kilpailijaan, VMware vSphereen. VSpherestä ja sen käytöstä oli aiemmin kuluvana vuonna Juuso Niemensyrjä kirjoittanut opinnäytetyönsä. Tämän perusteella Ovirtia vertailtiin vSphereen ja tehtiin pientä arviointia eroavaisuuksista ja yhtäläisyyksistä.

Yhtäläisyyksien osalta Ovirt ja vSphere ovat molemmat virtualisointialustoja, joilla on mahdollista ylläpitää useita virtuaalisia tietokoneita. Lisäksi ominaisuudet paperilla vaikuttavat erittäin samanlaisilta. Voikin sanoa, että Ovirt on rakennettu suoraksi kilpailijaksi VMwaren vSpheren rinnalle.

Suurin ero VMware vSpheren ja Ovirtin välillä on luonnollisesti lähdekoodin avoimuus. Koska Ovirt on avointa lähdekoodia, on se myös maksuton ja muokattavissa omaan käyttöön, jos taidot riittävät. Ovirtin kehitykseen on myös mahdollista osallistua ja yrittää vaikuttaa siihen, mihin suuntaan projekti lähtee etenemään jatkossa. Muina eroina Ovirtin ja vSpheren välillä ovat verkot, hallintaympäristö sekä virtualisoitujen koneiden hallinta.

Verkkojen osalta vSphere on Ovirtia hieman edellä. VSphereen on lisätty virtuaalikoneiden verkotusta varten virtuaalikytkimet, joiden avulla virtuaalikoneiden verkkorajapinnat voidaan eristää toisistaan (Niemensyrjä 2014, s.24). Verkkorajapintoihin on mahdollista sallia vain tietty liikenne. Myös Ovirtissä voi liikennettä rajoittaa ja ohjata verkkorajapintoihin, mutta varsinaisia virtuaalikytkimiä ei voi luoda. Uusia rajapintoja on mahdollista luoda, mutta rajapinnat ovat sidoksissa hostin fyysiseen sovittimeen (sovittimiin). Virtuaalikytkimet ovat Ovirtin sivujen mukaan kehityksen alla, mutta kirjoitushetkellä vielä siinä tilassa, että niitä ei ole integroitu kokonaisuuteen. Verkkorajapintoja voidaan muokata hallinnasta ja kuviossa 7 onkin näkyvillä perusasetukset yhdelle rajapinnalle.

Hallintaympäristö Ovirtin ja vSpheren välillä on hieman eroava. Siinä missä Ovirtin kanssa hallinta hoidetaan lähes tyystin web-käyttöliittymästä, ei vSpheren hallinta onnistu täydellisesti ilman erillistä ohjelmaa (Niemensyrjä 2014, s.16-17). Ohjelma tarvitsee pohjalle Windows-käyttöjärjestelmän, joten täysin puhtaaseen Linux-ympäristöön vSphere voi olla huonohko vaihtoehto. Puhtaassa Linux-ympäristössä ongelmaa voi yrittää kiertää virtualisoimalla Linux-työasemalle Windowsin, mutta tällöin hallinta tulee tehdä siihen varatulta työasemalta. Lisäksi virtualisoitu Windows tarvitsee lisenssin. Näin toimimalla myös kaikille vSphereä ylläpitäville henkilöille pitäisi virtualisoida yksi kappale haluttua Windows-käyttöjärjestelmää. Täten ratkaisua ei erityisen helppokäyttöiseksi tai edulliseksi voi sanoa.

Teoreettisesti Ovirtiin virtualisoituihin koneisiin on mahdollista ottaa yhteys selaimella, jos selaimeen saa asennettua tarvittut lisäosat. Riippuen asennusvaihtoehdoista virtuaalikoneessa, tulee ruudulle graafinen- tai terminaalikäyttöliittymä. Ominaisuus on kehitysvaiheessa ja merkitty hallinnassa ”experimental”-lisämääreellä. Tilaaajasta johtuvista Windows 7 -käyttöjärjestelmistä, ei ominaisuutta pystytty itse varmentamaan tai testaamaan, koska lisäosa ei suostunut asentumaan Internet Exploreriin. Firefox-selaimelle saatavaa lisäosaa ei ole vielä Windows-käyttöjärjestelmille ja Linuxien puoleltakin lisäosa oli saatavilla kirjoitushetkellä vain Red Hat, Fedora, CentOS ja Scientific Linuxille. Hallinta virtuaalikoneisiin onnistui kuitenkin avoimen lähdekoodin Virt-viewer-ohjelmalla, josta on myös Windowsille käännetty binääri. Palvelinkäytössä asiassa ei kuitenkaan ole merkitystä, koska usein Linux-palvelimen ylläpito hoidetaan terminaalista suojatun SSH-yhteyden ylitse. Windows-palvelimia taas ylläpidetään Windowsin omalla RDP asiakasohjelmalla (tai jollain avoimen lähdekoodin vastineella kuten rdesktop).

Vertailun päätteeksi voidaan todeta, että vSphere on hieman hiotumpi kokonaisuus verrattuna Ovirtiin ainakin vielä kirjoitushetkellä, mutta tulos ei juurikaan yllättänyt. VMware on ehtinyt kehittää ja panostaa vSphereen huomattavasti kauemmin kuin koko Ovirt on ollut edes olemassa.

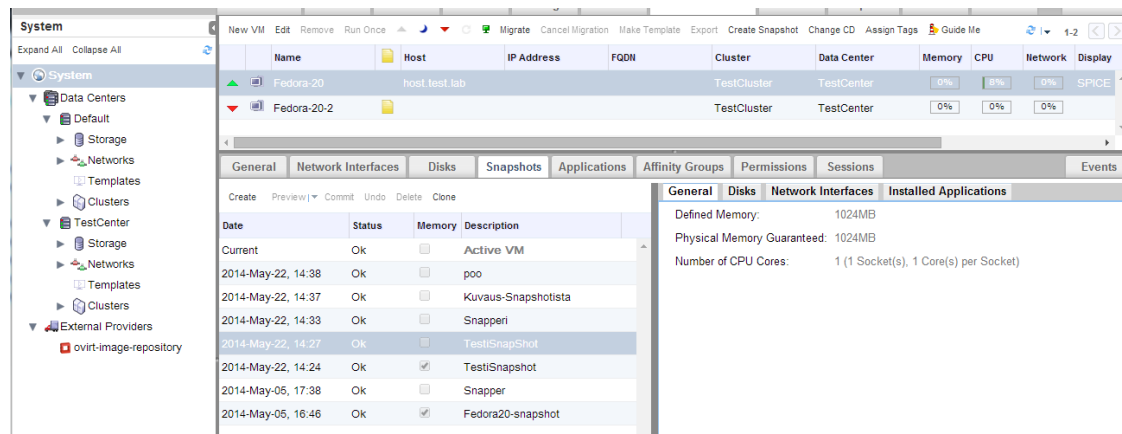
9 Päätelmät

Ovirt on jokseenkin toimiva kokonaisuus, mutta dokumentaation puute vaivaa lähes joka kulman takana. Ovirt-projektin käyttöönotossa onkin muutamia sudenkuoppia, joita ainakaan kirjoitus- ja tekohetkellä ei ollut dokumentoitu lainkaan Ovirtin omassa dokumentaatiossa. Tämä ei toisaalta ole erityisen yllättävää siihen nähden, että tämän kokoluokan projekti on vasta alkutaipaleellaan. Lisäksi asennuksen aikana ilmennyt pakettivarastojen osittainen puutteellisuus ei vakuuttanut siitä, että uusimman version käyttöönotto olisi tuotantoympäristössä mielekästä. Toki on mahdollista asentaa uusinta, mutta totuuden nimissä on sanottava, että asennus ei välttämättä ole aivan muutaman minuutin projekti ja ongelmien selvittäminen onkin pääsyy miksi käyttöönotto saattaa viivästyä.

Koska Ovirt on upstream-julkaisu Red Hatin Enterprise Virtualization alustalle, huomaa siellä täällä vikoja ripeän kehityksen seurauksena. Uudet ominaisuudet saattavat rikkoa jonkin vanhan toiminnon tai ne eivät toimi halutulla tavalla. Uusia ominaisuuksia ja korjauksia implementoidaan Ovirtiin hyvin tiuhaan tahtiin, joten ominaisuuksien hetkelliset viat ovat ymmärrettävissä. Useimpia ongelmia voi ratkoa joko itse, odottamalla päivitystä tai asentamalla suoraan versionumeroa pienemmän julkaisun, joista useimmat ongelmat ovat saatu poistettua. Näin toimimalla saavutetaan stabiilimpi tilanne joidenkin ominaisuuksien karsiutuessa pois.

Käytännön osuudessa Ovirtin kanssa tuli törmättyä myös ohjelmistovirheisiin, jotka saattoivat johtua Windows 7 -koneelle asennetusta Firefox-selaimesta, yhteensopivuusongelmasta tai Ovirtistä itsestään. Yksi virheistä ilmenee hallinnassa tehtyjen snapshottien osalta. Hallinta ei näyttänyt luotuja snapshotteja Firefoxilla, vaikka snapshotit olivat kyllä luotu lokitiedoston mukaan. Ongelmaan ei törmätty Google Chrome -selaimella.

”Sokkona”klikkailemalla aiemman snapshotin sai käynnistettyä niin, että virtuaalikone palautui aiempaan tilaan. Kuviossa 18 näkyvät snapshotit Google Chrome -internetselaimella, mutta Firefoxilla kyseinen kenttä oli tyhjä. Tästä löytyy vikaraportti Ovirtin bugtrackerista ja kyseinen vika tuntuu esiintyvän aina silloin tällöin, mutta ei aina.



Kuvio 18: Snapshotteja virtuaalikoneista vasemmassa alakulmassa

Käytännön osuudessa testattiin schedulingia, jossa Ovirtin pitäisi siirtää virtuaalinen kone toiselle hostille laitteistokuorman tasaamiseksi. Ominaisuus ei toiminut kuten oli odotettua ominaisuuden kuvauksen perusteella. Managerista käynnistettiin useampi virtuaalikone samalle isäntäkoneelle ja yhden virtuaalikoneen suoritinkäyttö nostettiin niin ylös kuin oli mahdollista. Suoritinkuorma nostettiin käyttämällä linux- ja unix-käyttöjärjestelmistä tuttua ”yes-komentoa. Komento kirjoittaa ilman lisäoptioita tekstiä yes toistuvasti terminaaliin (stdout) niin nopeasti kuin se on mahdollista. Komento putkitettiin vielä /dev/null ”tiedostoon”, siten että ainoastaan suoritin tekisi töitä, eikä mikään muu muodostuisi pullonkaulaksi testin aikana. Suoritinkäyttö nousikin nopeasti sataan prosenttiin. Hetken odottelun jälkeen toisen virtuaalikoneista olisi pitänyt siirtyä isäntäkoneelta toiselle, mutta näin ei käynyt kertaakaan. Kaikilla testikerroilla koko isäntäkone kaatui muutaman minuutin kuormituksen jälkeen ja Windows ilmoitti, että ohjelma on lakannut vastaamasta ja on suljettava. Kaatumista edelsi aina fyysisen tietokoneen jonkin komponentin hienoinen, mutta selvästi korvin kuultava naksuminen. Vika saattoi siis jäädä laitteistoon tai ongelma saattoi johtua sisäkkäisestä virtualisoinnista tai jopa jostakin VMware Workstation 9:n ominaisuudesta.

Opetuskäyttöön Ovirtin ottaminen on varmasti suhteellisen hyvä idea säästyneiden lisenssimaksujen myötä. Ovirt-virtualisointialustan pystytys opiskelijalle saattaa kuitenkin johtaa tilanteeseen, jossa käytettävissä ei olekaan minkäänlaista virtualisointiratkaisua. Ongelmapisteiksi muodostuvat Ovirtin tapauksessa jo pelkät laiteresurssit. Useammalle eri koneelle rakentuva Ovirt yksinkertaisesti suositelluilla tai minimilaitteistolla vie suhteellisen paljon

modernia laitteistoa. Tämä voikin johtaa tilanteeseen, jossa opintojaksolla ei ole riittävästi resursseja kaikille opiskelijoille. Lisäksi ohjelmistovirheiden ja nopeasta kehityksestä johtuvien puutteellisten pakettien aiheuttamat ongelmat saattavat hidastaa virtualisointialustan pystytystä huomattavasti. Lisäksi Ovirtin käyttöönotto voi olla hieman haastava operaatio dokumentoimattomien ominaisuuksien suhteen. Mahdollinen All-in-one-tyyppinen testiratkaisu, johon Ovirtin sivuilla on ohjeet tarjolla, saattaisi toimia opiskelijan käytössä, mutta tästä saattaa saada hieman vinoutuneen kuvan kokonaisuuden kannalta. All-in-one-ratkaisussa kaikki toiminnot hallinnasta isäntäkoneisiin toimivat yhden koneen päällä.

Käytännössä tuli myös huomattua, että Ovirt tarvitsee joko olemassa olevaa infraa tuekseen tai tätä on perustettava asennusvaiheessa. Käyttäjätilejä ei Ovirt managementille voi lisätä webhallinnasta laisinkaan ainakaan kirjoitushetkellä (sivujen mukaan tämä toiminnallisuus on rakenteilla), vaan erillisille käyttäjille täytyy käyttöoikeudet virtuaalikoneisiin ja niiden hallintaan tuoda jo olemassa olevalta autentikointipalvelimelta. Autentikointipalvelimina voi käyttää ainakin aiemmin mainittuja Windows Active Directorya tai FreeIPAA.

Autentikointipalvelimen lisäksi hommaa helpottaisi huomattavasti myös pilven ulkopuolinen nimipalvelin.

Etuina Ovirtillä esimerkiksi VMwaren-tuotteisiin verrattuna on sen erittäin edullinen hinta (0e). Koska VMwaren vSphere suurinpiirtein samoilla ominaisuuksilla kuin Ovirt on maksullinen, ei sitä saa suoraan ladattua ja asennettua testattavaksi. VSpheren kanssa suunnitelma toteutuksesta pitääkin olla valmiina ja 30-päivän kokeiluversio on rajoitettu ominaisuuksiltaan. Ovirt toisaalta on mahdollista ladata, asentaa ja kokeilla maksutta. Tämän lisäksi Ovirtin hallinnointi onnistuu internet-selaimella (vSpheren kanssa käytössä on Windows ohjelma hallinnointiin), joten pohjalle ei tarvita Windows käyttöjärjestelmää toisin kuin vSpheren kanssa käytännössä tarvitaan.

Koska virtualisointiin on useampaa erillistä työkalua ja ohjelmistoa saatavilla, on Ovirt vain yksi osa suuremmassa kokonaisuudessa. Avoimen lähdekoodin puolelta olisi mahdollisuus tutkia myös muita vaihtoehtoja suhteellisen helposti. Myös maksullisia kokonaisuuksia on saatavilla useita vaihtoehtoja. Avoimista tutkinnan alle on syytä laittaa ainakin Proxmox VE sekä Jyväskylän

ammattikorkeakoulussa FreeNest-projektissa käytetty OpenStack. Jyväskylän Ammattikorkeakoulussa on tehty muutama opinnäytetyö, jotka ainakin sivuavat OpenStackia ja sen käyttöä. Hakusanalla OpenStack löytyy Theseuksesta useampi eri työ, mutta kaikki töistä eivät keskity pelkkään OpenStackiin.

Kuten taustassa oli mainittuna, opinnäytetyötä kirjoittaessa erityisesti Ovirtiin liittyvän kirjallisuuden puute häiritsi kirjoitusta erittäin merkittävästi. Ovirtin oman verkkodokumentaation lisäksi piti tietoja kaivella Red Hatin dokumentaatiosta ja tämä aiheuttaakin pienehköä ristiriitaa ominaisuuksien välillä. Koska Ovirt on pykälän tai kaksi versionumeroltaan edellä ja ominaisuuksia on lisätty, ei kaikkeen Red Hatin dokumentaation voinut luottaa sokeasti, vaan tietoa piti hieman soveltaa. Työn tueksi olisi myös ollut saatavilla yksi julkaistu kirja, mutta koska kirjassa kerrottiin vanhemmasta Ovirtin versiosta, ei sen hankinta olisi ollut erityisen mielekäästä. Peruskonseptit kirjasta varmasti olisi tullut selviksi, mutta saman tiedon löytää myös Red Hatin dokumentaatiosta, joka onneksi on vapaasti jaossa ilman tukisopimuksia Red Hatin kanssa.

Lähteet

- Burden, A., Z. Dover, S. Gordon, T. Hildred ja D. Macpherson. 2014. ”Red Hat Enterprise Virtualization”. Viitattu 13.5.2014.
https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Virtualization/3.4-Beta/pdf/Technical_Guide/Red_Hat_Enterprise_Virtualization-3.4-Beta-Technical_Guide-en-US.pdf.
- Corbet, J., G. Kroah-Hartman ja A. McPherson. 2012. ”Linux Kernel Development”. Viitattu 6.5.2014.
<http://go.linuxfoundation.org/who-writes-linux-2012>.
- Day, M. 2012. ”KVM myths”. Viitattu 28.4.2014.
https://www.ibm.com/developerworks/community/blogs/ibmvirtualization/entry/kvm_myths_uncovering_the_truth_about_the_open_source_hypervisor?lang=en.
- IBM. 2007. ”Virtualization in education”. Viitattu 9.4.2014.
<http://www-07.ibm.com/solutions/in/education/download/Virtualization%20in%20Education.pdf>.
- Jyväskylän ammattikorkeakoulu. 2014. ”Tietoa JAMKista”. Viitattu 23.9.2014.
<http://www.jamk.fi/fi/Tietoa-JAMKista/>.
- Keegan, W. 2013. ”The Rise of the Type Zero Hypervisor”. Viitattu 28.4.2014.
<http://embeddedinnovator.com/2012/06/the-rise-of-the-type-zero-hypervisor/>.
- Milberg, K. 2009. ”IBM and HP virtualization”. Viitattu 28.5.2014.
<http://www.ibm.com/developerworks/aix/library/au-aixhpbvirtualization/au-aixhpbvirtualization-pdf.pdf>.
- Niemensyrjä, J. 2014. ”Työasemavirtualisoinnilla toteutettu palvelinvirtualisointi VMware vSphere -ohjelmistolla”. Opinnäytetyö, Jyväskylän Ammattikorkeakoulu, Tietotekniikka, Tekniikan ja liikenteen koulutusohjelma. Viitattu 9.5.2014.
<http://urn.fi/URN:NBN:fi:amk-201402262676>.

Ovirt. 2010. "Ovirt Documentation". Viitattu 9.4.2014.

<http://www.ovirt.org/Documentation>.

Ribeiro, S. M. 2009. "Virtualization Basics". Viitattu 12.5.2014.

<https://itechthoughts.wordpress.com/tag/protection-rings/>.

VMware. 2007. "Understanding Full Virtualization, Paravirtualization and Hardware Assist". Viitattu 28.5.2014.

http://www.vmware.com/files/pdf/VMware_paravirtualization.pdf.

Liitteet

Management

Ovirtin hallinta asennettiin Fedora 19 (Schrödinger's cat) GNU/Linux-jakelun päälle. Ensimmäisenä toimenpiteenä asennettiin levykuvalta ”minimal install”. Seuraavaksi jakeluun jo asentuneet paketit päivitettiin.

```
yum -y update
```

Lisäksi ladattiin ja asennettiin yksi irtopaketti, koska päivitysten myötä sos-paketissa (sos-2.2-31.fc19.noarch.rpm) oli vikaa. Aiemmillä asennusyrityksillä Ovirtin asennus keskeytyi ja epäonnistui pakettiristiriitaan.

```
curl -O ftp://fr2.rpmfind.net/linux/fedora/linux/releases/19/Everything/x86_64/os/Packages/s/sos-2.2-31.fc19.noarch.rpm && yum install sos-2.2-31.fc19.noarch.rpm
```

Lopuksi lisättiin Ovirt pakettisäiliö (repository) ja asennettiin Ovirt engine.

```
yum localinstall http://resources.ovirt.org/releases/ovirt-release.noarch.rpm
yum -y install ovirt-engine
engine-setup
```

Oletusasetukset Ovirt-hallinnan kanssa ovat hieman erikoiset siinä mielessä, että tekijät olettavat käyttäjän asentavan kaiken koneelle, jolla on graafinen käyttöliittymä. Tämä paljastuu siitä, että oletusasetuksilla Ovirtin asennuksessa asennusohjelma muokkaa iptables-sääntöjä niin, että koneen ulkopuolelta ei pääse hallintaan ilman, että sammuttaa palomuurin tai sallii TCP-liikenteen portteihin 80 ja 443. Tästä syystä muokattiin iptablesin sääntöjä ja sallittiin sekä portti 80 (HTTP) ja 443 (HTTPS) TCP-yhteyksille.

```
vi /etc/sysconfig/iptables
-A INPUT -p tcp --dport 80 -j ACCEPT
-A INPUT -p tcp --dport 443 -j ACCEPT
```

ISCSI-levysäiliö

Levysäiliökoneen käyttöjärjestelmäksi valikoitui Centos 6.5

GNU/Linux-käyttöjärjestelmä. Asennus tuli tehtyä turhan vauhdikkaasti, mutta suurempaa väliä tästä ei ollut. Liian vikkelän asennuksen myötä Centosille tuli asennettua graafinen työpöytä asennusskripteistä johtuen, vaikka sille ei ollut tarvetta. Suurempaa haittaa työpöydästä ei toki ollut, vain hienoinen virtuaalikoneen resurssien tarve.

Koneelle lisättiin ylimääräinen levy VMware Workstationilla, jonka jälkeen se alustettiin käyttövalmiiksi.

ISCSI-jaot lisättiin `/etc/tgt/targets.conf` tiedostoon, muutettiin hieman iptablesin sääntöjä `/etc/sysconfig/iptables` ja todettiin levyjaon toimivan.

Ovirt isännät

Ovirt-isännän (host-node) asentaminen ei ole erityisen iso operaatio. Ladataan vain levykuva Ovirtin sivuilta ja aloitetaan asennus.

Asennus ei syystä tai toisesta mennyt ”hiljaisena”(oletus) asennuksena koskaan loppuun, joten asennukselle annettiin kernel parametreinä ”hiljaisuuden” poisto ja muutettiin varmuuden vuoksi SELinux-tietoturvaominaisuus sallivaan tilaan. Toimenpide onnistuu painamalla asennusvalikossa sarkainta, pyyhkimällä ”quiet” pois ja lisäämällä loppuun rivi ”enforce=0”. Ilmeisesti hiljainen asennus pysähtyy samaan kohtaan kuin käsiasennus, jossa pääkäyttäjän (root) salasana jätetään tyhjäksi.

FreeIPA

FreeIPA-palvelin asennettiin Fedora 20:n päälle. Tässäkin asennuksessa tehtiin minimiasennus ja asennuksen päätteeksi asennettiin FreeIPA sekä poistettiin palomuuuri häiritsemästä.

```
yum install freeipa-server
systemctl disable firewalld
ipa-server-install
```

Asennuksen päätteeksi Windowsin Hosts-tiedostoa joudutaan muokkaamaan, koska FreeIPAn palvelin ohjaa liikenteen IP-osoitteesta nimiosoitteeseen. Ilman hosts-tiedoston muokkausta päädytään tilanteeseen, jossa DNS kyselyn epäonnistuesssa nimipalvelimelta puuttuvan tietueen vuoksi, ei Windows koneelta voinut hallita FreeIPAA. Hosts-tiedostoon määriteltiin FreeIPAn IP-osoite ja nimi johon sen pitäisi resolvata. Toinen vaihtoehto olisi ollut hallita FreeIPAA Fedoran omalla selaimella, mutta koska asennus tehtiin ilman graafista työpöytää, ei hallinta onnistunut. Hallintaa yritettiin Lynxillä (ascii-pohjainen selain), mutta se ei toiminut.

10.10.10.6 freeipa.test.lab

Lisäksi FreeIPAn asetustiedostoista jouduttiin muuttamaan yhtä riviä hieman. FreeIPAn asetustiedosto löytyy `/etc/dirsrv/slapd-TEST.LAB/dse.ldif` ja tänne muutettiin:

```
nsslapd-minssf=1 #normaalisti arvo on 0
```

Ovirt-management koneelle tehtiin seuraava käsky jonka jälkeen Ovirt autentikoi käyttäjät FreeIPAA vasten.

```
engine-manage-domains add --domain=test.lab --provider=ipa \
--user=admin --ldap-servers=freeipa.test.lab
```

Jostain syystä palvelu ei lähtenyt käyntiin joka kerralla virtuaalikoneen käynnistyessä, joten palvelua jouduttiin käynnistämään komentoriviltä uudelleen aina silloin tällöin.

GlusterFS

Gluster-levyjakoa varten otettiin käyttöön kaksi virtuaalikonetta ja määritettiin nämä jakamaan levyjään. Molemmat koneet asennettiin Fedora 20 -jakelulla minimiasennuksena. Asennuksessa määriteltiin molemmille koneille kaksi kiintolevyä, joista toisia kiintolevyjä käytettiin Gluster brickeinä.

Molemmille koneille asennettiin system storage manager.


```
yum install system-storage-manager
```

Asennuksen päätteeksi SSM:llä otettiin käyttöön toiset kiintolevyt Glusteria varten ja levyille valittiin tiedostojärjestelmäksi XFS. Lisäksi tehtiin valmiiksi hakemisto mounttaamista varten.

```
ssm add -p GlusterPool /dev/sdb
ssm create -p GlusterPool --fstype xfs -n gluster
mkdir /gluster
```

Koska levyt haluttiin myös käyttöön uudelleenkäynnistyksessä, joudutaan ne lisäämään fstab-tiedostoon. Seuraavissa kohdissa levyistä otetaan UUID (universally unique identifier) ja lisätään se /etc/fstab-tiedostoon.

```
blkid /dev/pool/gluster
```

Nyt fstabin pitäisi näyttää jotakuinkin tältä:

```
UUID=uniikki-id /gluster xfs defaults 1 2
```

Fstabiin lisäyksen jälkeen mountataan levyt (tai käynnistetään kone uudelleen):

```
mount -a
```

Palomuuuri poistettiin käytöstä, koska palomuurin kanssa ei haluttu taistella tässä vaiheessa työtä. Glusteria varten on ylläolevissa kohdissa luotu levyt käytettäväksi. Seuraavissa kohdissa lisätään Glusterfs-server, muokataan sen asetuksia, käynnistetään palvelu ja luodaan hajautettu levy. Ovirtia varten on muokattava /etc/glusterfs/glusterd.vol asetuksia niin, että option base-port kohdalta poistetaan kommentti (# -merkki) ja vaihdetaan portti 49125 portiksi 50152, jotta libvirtin kanssa ei tule ongelmia live-migrationin kanssa.

```
yum -y install glusterfs-server
service glusterd start && chkconfig glusterd on
```

Lisäksi vain toiselta koneelta tarvitsee seuraavan komennon:

```
gluster peer probe (toisen koneen ip tai hostname)
```

Lopuksi lisättiin vain toisella koneella Glusteriin jaettavaksi tarkoitettut levyt, annettiin oikeudet VDSM:lle ja käynnistettiin jako.

```
gluster volume create data replica 2 $GlusterKone1-IP:/gluster/data0 \
$GlusterKone2-IP:/gluster/data1
```

```
gluster volume set data storage.owner-uid 36 && gluster volume set \
data storage.owner-gid 36
```

```
gluster volume set storage group virt
```

```
gluster volume start data
```

Ovirtin käyttö

Kirjautumisen jälkeen pitää ensimmäisenä määrittää data center. Tämä onnistuu lisäämällä managementtiin uusi data center. Data centerin lisäämisen jälkeen tarvitsee Ovirtiin lisätä isäntäkone. Isäntäkoneen lisäämisen jälkeen data centeriin voi lisätä levytilan. Jos data centeriin lisää useamman isäntäkoneen, joudutaan levyn liitospisteeksi valitsemaan jokin isäntäkoneista. Kyseisestä isäntäkoneesta tulee välittömästi data centerin SPM (storage pool manager). Uudelleenkäynnistysten jälkeen isäntäkoneet pitävät vaalit keskenään kenestä tehdään SPM. Kun koneet ovat päättäneet, kuka on SPM, voi hosteille ruveta lisäämään virtuaalikoneita.

Virtuaalikone välilehdeltä voi lisätä uudet virtuaalikoneet halutuun määritykseen. Virtuaalikoneelle on hyvä lisätä cd-rom asema ainakin ensimmäiseksi käynnistyskerraksi, jotta virtuaalikoneen käyttöjärjestelmän voi asentaa. Asennus kestää hetken, mutta sopivilla levykuvilla sen voi automatisoida täysin.

Virtuaalikoneita voi tarkastella virt-viewer-ohjelmalla tai yrittää käyttää SPICE/HTML5 liitännäisiä. Liitännäiset kirjoitushetkellä tarvitsevat tosin käyttöjärjestelmäksi jonkin GNU/Linuxin, selaimeksi Firefoxin sekä SPICE liitännäisten asennuksen (pakettivarastoista liitännäiset on tarjolla ainakin Red Hatille, Fedoralle, Centosille ja Scientific linuxille). Toinen vaihtoehto on

Windows, Internet explorer ja activex liitännäinen, mutta tämä ei toiminut työn suorituksen aikana.

Isäntäkoneen rikkoutuessa (VMware Workstationilta virtojenkatkaisu) Ovirtiltä kestää jokunen minuutti huomata että isäntäkone ei enää toimi. Tämän jälkeen virtuaalikoneen pitäisi siirtyä automaattisesti toiselle hostille, mutta niin ei tunnu aina käyvän. Lisäksi sammuttaessa SPM-isäntäkoneen, muuttuu koko data center toimimattomaksi (hetkellisesti). Toisaalta taas poistamalla verkkosovittimelta IP-osoitteen poistui hostilta ovirtmgmt verkko ja virtuaalikone siirtyi välittömästi koneelta toiselle.

Verkkojen pitäisi toimia periaatteessa niin, että isäntäkoneet tunnistavat omat verkkokorttinsa. Näiden päälle voi rakentaa VLAN-tägättyjä verkkoja.

Esimerkiksi SPICE protokollan (virtuaalikoneen hallinta managerista) voi erottaa omaan VLANiinsa, jolloin kyseinen liikenne kulkee ainoastaan sen verkon lävitse. Myös levytilan liikenteen voi erottaa omaan verkkoonsa ja tämä tuntuukin hyvältä idealta tuotannon osalta, mutta verkkojen erottamista ei ole pakollista tehdä.